Author:
Fletcher, P

Title:
Truth, proof and infinity.

Truth, Proof and Infinity

by Peter Fletcher

A thesis submitted to the University of Bristol in accordance with the requirements for the degree of Doctor of Philosophy in the Faculty of Science.

School of Mathematics, June 1988.

Title: Truth, Proof and Infinity, Author: Peter Fletcher,

PhD thesis, University of Bristol, June 1988.

Abstract

This thesis is an attempt to provide an adequate foundation for mathematics along roughly intuitionistic lines.

I criticise set-theoretic foundations and develop an alternative philosophy firmly rooted in constructive mathematical experience. I also discuss the role of formal systems in a global account of mathematics.

Intuitionism presupposes a 'theory of constructions' (a 'protologic') underlying logic and mathematics. Past attempts to supply such a theory have failed to clarify its philosophical nature or role in mathematics. I give my own protologic, and use it to interpret intuitionistic logic, Heyting Arithmetic, and classical analysis.

Title: Truth, Proof and Infinity, Author: Peter Fletcher,

PhD thesis, University of Bristol, June 1988.

Abstract

This thesis is an attempt to provide an adequate foundation for mathematics along roughly intuitionistic lines.

I criticise set-theoretic foundations and develop an alternative philosophy firmly rooted in constructive mathematical experience. I also discuss the role of formal systems in a global account of mathematics.

Intuitionism presupposes a 'theory of constructions' (a 'protologic') underlying logic and mathematics. Past attempts to supply such a theory have failed to clarify its philosophical nature or role in mathematics. I give my own protologic, and use it to interpret intuitionistic logic, Heyting Arithmetic, and classical analysis.

Acknowledgement: I have benefited from many discussions with Dr J.P.Mayberry during the research on this thesis.

Declaration: This thesis is all my own work.

Peter Fletcher

Errata

Page 59, line 19: 'The Recursion Theorem' should read, 'The Second Recursion Theorem'.

Page 59, lines 22-24: Omit the sentence beginning, 'The g so obtained ... '.

Page 68, lines 6-15: Replace the passage from 'A function fxpt Φ... ' to the end of the paragraph by

'From the usual proof of the Second Recursion Theorem it follows that $fxpt\,\Phi(\underline{x})$ does not merely *equal* $\Phi(fxpt\,\Phi)(\underline{x})$: $fxpt\,\Phi(\underline{x})$ is evaluated by constructing the term $\Phi(fxpt\,\Phi)(\underline{x})$ and evaluating it. Therefore, if Φ is of the form specified in the Fxpt Rules, $fxpt\,\Phi$ works by applying H to its argument $\underline{x}$ repeatedly until it satisfies C, then applying R to give the final result. This makes $fxpt\,\Phi$ the *least* fixed point of Φ (for this class of Φ): then $fxpt\,\Phi = \bigcup_n \Phi^n(\mathcal{U})$, where $\mathcal{U}$ is a nowhere-defined function [13,Chapter 4].

Fxpt Rule (a) may be informally justified as follows. The conclusion, $fxpt\,\Phi(\underline{x}) \to Y(\underline{x})$ (omitting the initial variables for brevity), follows from $\forall n(\Phi^n\mathcal{U}(\underline{x}) \to Y(\underline{x}))$. This is proved by induction on n. The basis case is $\mathcal{U}(\underline{x}) \to Y(\underline{x})$, which is trivial. The induction step is as follows: by applying Φ to the inductive hypothesis we obtain $\Phi^{n+1}\mathcal{U}(\underline{x}) \to \Phi Y(\underline{x})$, which $\to Y(\underline{x})$ (by the premise), as required.

Fxpt Rule (b) is proved similarly. $X(fxpt\,\Phi_1,\ldots,fxpt\,\Phi_k)(\underline{x})$ is $X(\bigcup_n \Phi_1^n\mathcal{U},\ldots\bigcup_n \Phi_k^n\mathcal{U})(\underline{x})$, which is $\bigcup_n X(\Phi_1^n\mathcal{U},\ldots\Phi_k^n\mathcal{U})(\underline{x})$: we need to prove that this $\to fxpt\,\Psi(\underline{x})$. This follows from $\forall n(X(\Phi_1^n\mathcal{U},\ldots\Phi_k^n\mathcal{U})(\underline{x}) \to fxpt\,\Psi(\underline{x}))$, which is proved by induction on n. The basis case is trivial. The induction step is $X(\Phi_1^{n+1}\mathcal{U},\ldots,\Phi_k^{n+1}\mathcal{U})(\underline{x}) \to \Psi X(\Phi_1^n\mathcal{U},\ldots\Phi_k^n\mathcal{U})(\underline{x})$ (by the premise), which $\to \Psi(fxpt\,\Psi)(\underline{x})$ (by applying Ψ to the inductive hypothesis), which is $fxpt\,\Psi(\underline{x})$, as required.

Note that these arguments only work because of the highly restricted form of X and Φ, Ψ, ... They are intended to justify the Fxpt Rules informally to someone who accepts numeric induction; however, I regard the Fxpt Rules as more primitive than numeric induction.'

Contents

Chapter 0: Philosophical Remarks

§0.0 What's wrong with set theory?

Classical set theory is an attempt to 'actualise' potential infinities. Thinking of, say, the set of natural numbers $\mathbb{N}$ as if it were a heap of physical objects, classical logic assumes that $\forall x \in \mathbb{N}$ and $\exists x \in \mathbb{N}$ are legitimate sentence constructs: that is, we can insert either in front of a well-defined concept $\Phi(x,y,\ldots z)$ to obtain another well-defined concept $\Psi(y,\ldots z)$. Similarly with any other infinity in place of $\mathbb{N}$.

I think the reason mathematicians believe this (at least, the reason I used to believe it) is that since we have a clear idea of the concepts that generate the 'potential infinity' $\mathbb{N}$ (zero and the successor function) this determines uniquely the 'actual' $\mathbb{N}$. There may be philosophical qualms about its 'existence' (whatever that means), but, since it is unambiguously determined, quantified statements about it have objective truth-values, and there seems to be no obstacle to deducing some of them by predicate calculus, leaving philosophical questions to be settled later (or never).

Of course, the most this argument claims is that there is at most one actual $\mathbb{N}$. But it is questionable whether it even succeeds in that: Non-Standard Analysis shows that it is perfectly consistent to imagine two actual $\mathbb{N}$'s each interpretable as an actualisation of the progression $0,1,2,3,\ldots$.

The main problem, however, with set theory is that it gives rise to a new potential infinity (the universe of sets, V), which, although it is usually assumed we can quantify classically over it, is not a proper object in the way that individual sets are. Hence the basic dogma of set theory: all 'collections' are either sets or non-sets. This distinction is based on the ancient view that only collections 'limited in size' are permissible (eg Eudoxus' definition of a number as a limited plurality): only they form 'organic wholes' or 'consistent multiplicities' (which is nonsensical: only *theories* can be consistent) or 'collections that can be thought of as one' (although in saying 'a collection' (a singular noun phrase) we are already thinking of it as one). Of course, if you had asked an ancient or medieval logician what 'limited in size' meant he would have explained that the limitedness consisted in the fact that one can obtain a strictly larger collection by adding an extra element; he might have added that the limited whole is strictly larger than any of its proper parts. These principles have to be abandoned when Cantor interprets limitedness in a broader sense than ordinary finiteness, to make $\mathbb{N}$ and $\mathbb{R}$ 'limited', so that there is not much left of the original notion.

Cantor believed in the limitation of size idea for theological reasons: sets are collections directly imaginable by God (whereas we can only imagine collections finite in the ordinary sense) while V ('Absolute Infinity') is God-sized. We can talk about sets freely because they exist as objects in God's mind, but we cannot talk about V. This view doesn't work even in its own terms (cf Hallett [7,Chapter 1]). Modern set theorists maintain the distinction simply to avoid the paradoxes. Still, I suspect that quasi-theological ideas linger, accounting for why the idea feels right to mathematicians. The ordinal hierarchy is the set theorist's Tower of Babel, and if we try to use it to climb to Absolute Infinity we must expect to be struck down by contradictions.

The sole advantage of this approach is that it treats $\mathbb{N}$ and the other collections useful in practical mathematics as limited, so that they can be manipulated almost as if they were finite. The disadvantages are as follows.

Because 'limitedness' is obtained from the ordinary notion of finiteness, set theory is haunted by the ghost of constructivity: I mean the view of V as a cumulative hierarchy obtained from the empty set by 'iterating' the power set and replacement operations transfinitely. This makes me wonder whether the Zermelo-Fraenkel axioms (ZF) are even consistent. As far as I know, there are only two arguments for consistency: that the axioms embody coherent informal ideas (but they don't: they are a mixture of generative/quasi-constructive and static/impredicative views), and that the whole of mathematics may be regarded as an unsuccessful attempt to deduce a contradiction from the axioms (but this is unconvincing: practical mathematics rarely uses the full impredicative strength of the replacement and separation schemata, and only quantifies over V in fairly innocuous ways).

The fact that only sets can be actual infinities undermines the motivation for introducing actual infinities in the first place: for any argument that justifies going from potential $\mathbb{N}$ to actual $\mathbb{N}$ also justifies going from potential V to actual V.

Moreover, it all seems remote from mathematical experience. There are transfinite 'constructions' that can only be carried out by God, and a huge multitude of previously unsuspected and apparently useless sets, most of them undefinable; whereas we are prevented from defining things we need (proper classes). The fact that all questions about cardinals except the simplest are undecidable from the axioms has led many set theorists to deny that we are talking about 'the' universe of limited multiplicities at all, and speak instead of alternative set theories, apparently with no notion of an intended model. The impression of arbitrariness about the properties of V is unsurprising when we consider that it is based on a notion of limitedness which has been detached from its original context (ordinary finiteness) and left hanging in mid-air; and the power set axiom blindly posits the existence of the power set of any set without telling us anything about it beyond its bare existence (it is no good saying that the power set is characterised extensionally, and hence completely, because the power set operation itself is used to decide what is a set, via the cumulative hierarchy picture). Our philosophy of mathematics should be embedded in general philosophy, so its concepts should be somehow 'grounded' in our basic experience. It is surely implausible to say: "Mathematics is that branch of our knowledge that deals with the irreducible concept of *limitedness*".

While set theorists endlessly dispute how many super-huge cardinals can dance on the head of a pin, practical mathematics either sticks to the first few levels of the cumulative hierarchy or talks about completely arbitrary structures (which could be classes, families of classes, or anything at all): it seems to pay no attention to the set/non-set distinction. For example, theorems of group theory still apply when the group is class-sized; and the 'category' of all categories satisfies the usual category-theoretic theorems.

Set theorists sometimes justify their enormous but limited universe by pointing to theorems relating large cardinal hypotheses to the highly impredicative properties of the continuum. (In the language of Ramified Type Theory, high type concepts are being related to high order ones.)

This is supposed to show that large cardinal problems are 'real' questions. What it actually shows is that the impredicative continuum and large cardinals stand or fall together; both seem equally irrelevant to the part of mathematics that relates to physical applications. The classical continuum is supposed to be the structure of space-time, at least locally, yet can anyone suggest an empirical test for the continuum hypothesis, Martin's axiom, the axiom of determinacy, or the hypothesis that all Borel subsets of the reals are Lebesgue-measurable? This suggests a serious mismatch between the classical continuum and physical space-time (which may not be a set of points at all): if the continuum hypothesis is either true or false when stated of physical space-time but its truth-value is unobservable then we are not talking about physics but metaphysics. ZF is misleading in giving the impression that whereas large cardinals are exotic the properties of the reals are not because they are low down in the cumulative hierarchy.

ZF set theory has established itself as the orthodox foundational system because, unlike its rivals, its inadequacies are such as to allow mainstream mathematics to continue as before, at the cost of losing touch with the motivating ideas behind mathematics. My main complaint is that, due to its reliance on the obscure notion of limitedness, it bungles the problem of how to talk about the whole universe: it lumbers us with unwanted sets and prohibits apparently meaningful definitions (the universal set, the set of ordinals).

§0.1 Criteria for a foundational system

(α) An account of the foundations of mathematics should be founded on our immediate experience of mathematical activity (eg counting, computations, reasoning about computations); it should not begin by simply postulating axioms which 'force themselves upon us as being true' about an unexplained universe of mathematical objects. In other words, it should not take 'existence' of mathematical 'objects' as an unanalysed primitive notion.

(β) It should explain how, despite (α), mathematical statements are not translatable into descriptions of the contents of particular mathematicians' minds at particular times, but are necessary, timeless and independent of the thinking subject (in the sense that they would still be true even if no mathematicians had ever existed).

(γ) It should justify as much as possible of existing mathematics (possibly reinterpreting it, since all mathematicians are implicitly applying a naive philosophy of mathematics).

(δ) It should not make stronger ontological assumptions than are implicit in 'practical' mathematics (I mean, roughly, mathematics which is not motivated by foundational considerations); since the foundational system is partly justified by saying that it is necessary for doing mathematics, which seems to be a legitimate activity. Set-theoretic concepts seem incapable of justifying existing arithmetic and analysis without introducing at the same time strong impredicativity and a vast cumulative hierarchy which dwarfs what had previously been considered the whole of mathematics. These are not added bonuses of the set-theoretic approach: they are (possible) defects. A theory which justifies arithmetic and analysis without such a large hierarchy would be preferable.

(ϵ) It should explain how and why mathematics is applicable to science.

§0.2 What's wrong with set theory? (continued)

In this section I shall continue the criticism of the 'platonist' approach to mathematics, but not focusing specifically on ZF, and with a view to diagnosing and curing the underlying ills.

The key problem is: is the universe of mathematical objects itself a mathematical object? For example, take the formalist approach. I have no quarrel with the basic formalist thesis that mathematics is the study of formal systems: it leaves untouched the real problem, namely what formal systems are possible and what we can say about them as a totality.

Foundational systems generally are inspired by certain modes of reasoning or formal expression; they posit a universe of objects (sets, formal systems, or whatever) and assume it closed under some 'extension principles' for generating new objects from given ones.

For example, in ZF the power set, pairing, union, separation and replacement axioms all have the basic form $\forall x \exists y\ \Phi(x,y)$ where, given x, $\Phi(x,y)$ determines y uniquely (except that for pairing there are two variables in place of x and for replacement the basic form is conditional on an antecedent). What these axioms are really doing is justifying a definition ('construction', almost) of y in terms of x.

They should not be regarded as asserting the existence somewhere of a y for each x but as legitimising certain 'constructions' we think we need. The term 'construction' raises questions about the existence of mathematical objects, which I shall deal with in §0.3. My point here is that our motivation for, and hence our reason for believing, these axioms depends on a quasi-constructive viewpoint.

The problem is that we need to talk about the whole universe in developing the system; moreover the universe is supposedly a meaningful concept (eg, 'set' is supposedly meaningful: that is, we claim we know what we are talking about) so we ought to be able to talk about it; but the universe is (in most systems) not an object within itself. At this point the temptation is to forget what we are trying to do and merely study the formal system in front of us. If we retain a dim memory of our original intention we may defend the system by claiming that it is adequate for present-day mathematics and if we ever find it inadequate we can enlarge it by including plausible new extension principles. This is probably true, but the questions remain of what extension principles are legitimate and whether arguments about 'arbitrary' objects remain valid when the universe is enlarged by new extension principles: this is essentially the problem we started with.

One possible view is that the universe is not a mathematical object; that is, it is not a legitimate object of mathematical thought; that is, we are not entitled to talk or think about it at all as mathematicians. This implies a stratified universe: every object has an implicit limited range of significance (its 'type'). The trouble with this view is that it cannot express its central dogma, for 'every object' in the preceding sentence is meaningless. It is no good invoking type ambiguity: if 'every object has a type' is type-ambiguous then it means 'every object of type τ has a type' (regardless of τ), which says nothing. The type theorist is in the position of an ancient Hebrew theologian, forbidden to pronounce the name of his god. The notion of type ambiguity itself is unexplainable because we need to understand that τ (in the above sentence) is itself of ambiguous type; type ambiguity has to be treated as a primitive notion, which is tantamount to assuming a primitive ability to assert a formula $\Phi(x)$ 'for arbitrary x'.

In practice type theorists do talk about types and objects generally in a way that cannot be expressed in the system (which is supposed to be all of mathematics); so they call such talk 'meta-mathematics'. But the 'meta' prefix accomplishes nothing. For the practising mathematician, who doesn't want to be bothered with type restrictions, is liable to respond, "Very well, by your terminology I am not a mathematician but a meta-mathematician: when I speak of *five* I mean the metatheoretic *five* (as in 'five types'), not any of the internal *fives* (there is one for each type). Now please develop a philosophy of meta-mathematics to justify what I do."

Such a justification must be meta-meta-mathematics, which must be justified in turn. We can spot the obvious pattern, and so define meta^n-mathematics as a function of n; the function itself is not in any meta^n-mathematics, so it must be an element of what we could call 'meta$^\omega$-mathematics'. In this way we can get meta$^\alpha$-mathematics for any constructive ordinal α, and all of them together make meta$^{\omega_1}$-mathematics (where ω_1 is the least non-constructive ordinal), and so on. Here the question arises: do we 'really' believe in non-constructive ordinals, or even 'all' constructive ordinals (not just as theorems of ZF)? This is the kind of question we were trying to avoid by setting up a formal system in the first place.

The reason for the difficulty is that I am seeking an account of the kinds of concepts legitimately definable in the course of doing mathematics; whereas most people working in foundations seem to regard this as hopelessly overambitious and instead want to develop a limited system adequate for present-day mathematics (classical or intuitionistic). There is no contradiction between our views; we are merely doing different things. I am studying the philosophy of mathematics whereas they are *doing* mathematics.

I have discussed the Cantorian view of set theory; there is an alternative Frege-Russell approach which sees sets or classes as extensionalised predicates, and explains the paradoxes by the vicious circle principle. This seems far more plausible to me, as it explains the fundamental objects of the system in terms of predicates which we define, and hence from meaningful mathematical activity, instead of postulating a mysterious objective mathematical 'world' of sets.

The central error of classical mathematics is its view of mathematics as a world separate from ourselves which we discover facts about. Classical mathematics breaks down for roughly the same reasons as classical physics does: the world is inextricably mixed up with our thoughts about it. As soon as we think we know what the mathematical universe is we can talk about it as an entity, and so transcend it. The problem with infinity is not that it is too large, that we are biting off more than we can chew, but that it refers ultimately to what we can count up to or define, and hence we are talking about our own possible thoughts; so references to the whole universe, or even any infinite domain, may be circular, and sentences containing unbounded quantifiers are not manifestly well-defined.

We have in fact no automatic right to use unbounded quantifiers at all: a formula containing them is devoid of meaning once the classical picture of a mathematical world outside ourselves is abandoned. (Whether quantification over a smaller, but still infinite, domain means anything remains open.) Yet we have an irresistible urge to use such quantifiers anyway, and mathematics would be impossible without them.

A particularly innocuous form of unbounded quantification is $\forall x\ A(x)$ where $A(x)$ is quantifier-free and $\forall x\ A(x)$ is proved by proving $A(x)$ 'regardless of x'. (Russell [18,p.158] singles these propositions out: he says that $A(x)$ is proved 'for any x' rather than 'for all x'.) The proof is an *argument schema*, with *schematic variable* x, leading to $A(x)$; the schema is recognised as a valid argument without needing to inquire into the value of x. Here I am taking an intensional view of arguments: an argument is valid, not merely if the conclusion is true whenever the premises are true, but iff the reasoning process the argument describes is correct; that reasoning process need not use all the information known, in particular it need not use the value of x. Any argument obtained by substituting a name N of any object for x in the schema (or equivalently adding the information $x=N$) will also be valid. The preceding sentence seems to quantify over objects, but that does not mean that I am now admitting 'for any object, ...' as a valid sentence construct; the sentence is meaningful because it is a sort of prediction, or rather a *licence* (and so strictly speaking not a proposition) to regard arguments of a certain form as valid in future without needing to examine them afresh. Generally, 'for all x, $A(x)$' is not directly meaningful, even if $A(x)$ is understood: it needs to be given a 'predictive', experiential meaning.

Call this very restricted version of quantificational logic <u>protologic</u>.

Yet it is difficult to stop at this point. One wants to say, eg, that if a function f maps everything to a natural number then so does the function Sof obtained by composition with the successor function. This statement is of the form

$$\forall x\, A(x) \supset \forall x\, B(x), \qquad \text{(i)}$$

with $A(x)$ and $B(x)$ quantifier-free. The obvious argument for (i) actually establishes

$$\forall x\, (A(x) \supset B(x)), \qquad \text{(ii)}$$

and this is protologically meaningful. The natural way of understanding (i) is as an 'incomplete' version of (ii), where we have abstracted one 'aspect' of (ii), namely the fact that it enables us to pass from $\forall x\, A(x)$ to $\forall x\, B(x)$ (both of which are protologically meaningful).

Clearly we could have obtained (i) as an incomplete version of, say,

$$\forall x\, (A(gx) \supset B(x)),$$

or, more generally, any protological argument allowing us to infer $\forall x\, B(x)$ from $\forall x\, A(x)$.

Having admitted incomplete statements, it is then easy to interpret $\exists x\, A(x)$ as an incomplete version of $A(a)$, in which we have abstracted one 'aspect' of $A(a)$, namely that it asserts that A holds of something.

Clearly we are talking about intuitionistic logic, based on a protologic of statements true 'regardless' of the values of their schematic variables. This suggests a programme for interpreting mathematics, which I shall carry out in subsequent chapters: I define protologic in Chapter 1, apply it to arithmetic in Chapter 2, and apply it to analysis in Chapter 3.

The controversy over intuitionistic logic has always concentrated on why we can't use excluded-middle arguments; we should have been asking, what makes us think we can quantify at all? General predicate-logic formulae cannot be interpreted as propositions. Only Π_1 statements can be given a direct meaning (and then not the full classical one); other formulae can be given an indirect meaning which justifies almost all the arguments we are used to using in predicate logic. This is far more than we have any right to expect.

Restricting ourselves to intuitionistic quantifiers gives some hope of talking about the universe without presupposing the classical picture. But the central problem remains: whenever we have a domain of objects we can transcend it by talking about the domain itself. This 'open-endedness' is the central fact of mathematical life: philosophy of mathematics should give an account of it, not deny it by confining us to a fixed formal system on pain of contradiction.

§0.3 Idealisation and the geometry of time

In this section I begin the attempt to formulate an adequate foundation for mathematics. Start with Criterion (α) of §0.1. (Pure) mathematics is based on idealised arguments about our counting and computational experiences. The subject matter is the manipulation of expressions in a finite (ie explicitly listed) alphabet; any repeatable action which yields a new result after each repetition may be regarded as a system of numerals (eg drawing chalk marks on a blackboard), and we are concerned with manipulations of these according to rules expressed in a finite alphabet (ultimately codable as numerals).

This view of mathematics is properly describable as formalist. I mean by formalism the interpretation of mathematics as being concerned with the manipulation of formal strings of symbols: the strings are intrinsically meaningless, but the fact of being able to obtain a string by specified rules has a clear 'finitistic' meaning, and can be used to deduce computational statements by 'finitary' reasoning. The fact that the manipulation rules are called 'rules of inference', the starting strings are called 'axioms', and the manipulations are reminiscent of stylised reasoning processes, is surely inessential since the strings are not interpreted as propositions. (I call mathematics roughly what Hilbert calls metamathematics.)

It may be objected that symbols are abstractions: all we really have are the written tokens on the page, or whatever else we are using to record symbols, and these are unsuitable in that there are only finitely many of them, they may be imperfectly formed, we may misread an "α" as an "a", etc. Granted, but the fact that we understand what it means to misread a token shows that there is a clear notion of 'a token intended to be of type "α"'. When we express ourselves in a finite alphabet we are chopping our thoughts up into small sharply-defined pieces: I take these pieces as the building blocks of mathematics.

Finiteness will be elucidated in terms of iteration or what we can count up to, since counting is immediately given as an algorithm. Having obtained finiteness we can then look back and view iteration as having a 'finite' and a 'temporal' aspect (ie as repeating a process 'finitely many times'); it might then appear that finiteness is more primitive than iteration and that viewing finiteness via iteration is pointless. But the fact remains that, in the way I am doing things, iteration comes *first*, in our ability to execute algorithms (I think this is related to Brouwer's primal intuition), and without it we would never get as far as a notion of finiteness from the perspective of which we could look down on iteration.

Now, mathematics only considers certain aspects of this token-manipulating activity. Suppose we regard sequences of chalk marks drawn on a blackboard as stroke numerals representing numbers. Then it is true to assert:

(i) We cannot write down an even prime number > 10.

(ii) We cannot write down an odd prime number > 10^{10}.

Informally, there is a clear difference between the reasons for (i) and (ii); (ii) is due to the fact that we will run out of chalk, blackboard space and patience long before we can write down the number; (i) seems to reflect essential 'structural' properties of counting processes which would still hold if we represented numbers by counting sheep entering a pen or electrons passing one at a time between two electrodes (for which (ii) might be false).

Arithmetic seems to me to be an attempt to talk about an unspecified process carried out repeatedly, disregarding contingent physical limitations (running out of chalk, etc.) peculiar to particular processes. A <u>number</u> is a stage in this anonymous iteration.

More generally, we want to 'introduce' (or 'construct' or name) <u>mathematical objects</u> by giving their computational meaning: a function is given by explaining how to calculate its value for an arbitrary argument, a set S by explaining the computation $x \in S$, a formal system by specifying an algorithm for checking proofs, and so on.

This procedure relies heavily on abstraction and idealisation. Mention of 'abstract objects' raises the questions: what exactly *are* abstract objects? what does it mean to argue about their existence? are they just figures of speech? do definitions *create* objects or simply *name* them? does every definition of a function (by explaining how to calculate it) define an object? are all abstract objects obtainable (or nameable) in this way? does idealisation change the properties of a real object to make it ideal, or create an ideal object alongside the real one, or neither?

The metaphorical view of mathematics as the natural history of an inaccessible world of abstract objects is useful, but a little preposterous. Suppose I try to define a function $f:\mathbb{N} \to \mathbb{N}$ by

$$f0 = 1,$$
$$n > 0 \Rightarrow fn = n \times f(n-1).$$

Am I obliged to justify my definition by switching on my Gödelian set-theoretic intuition (assuming I have one) and rummaging through the mathematical universe, examining objects one at a time and piling them in a heap, in the hope that when I have exhausted the universe I will have found one, and only one, object satisfying the defining equations? Or should I employ a specialised search-and-rescue service such as the ZF 'recursion theorem', guaranteed to locate a unique referent for any primitive recursive definition?

I refuse to do anything of the sort. When I talk about the existence of a function I want to mean simply that I know how to calculate it. The function should *be* an algorithm; the understanding of its computation should automatically justify it and exhaust its meaning.

To have an independent and prior concept of 'existence' simply falsifies the intentions behind mathematics.

I have said what I want, but how do I get it? In particular, what is the idealisation process involved? what do statements about ideal objects *mean* (are they all translatable to statements about real objects)? are such statements literally true? and if not, can they still be justified as 'useful'?

Taking up the last question, there is no doubt of the usefulness, even the necessity, of idealisations in ordinary arguments, inside and outside mathematics. All our knowledge about the world consists of idealisations in which inconvenient aspects of a situation are neglected, either because we do not know them, or we are confident they would make little difference to the answer, or for generality, or to simplify argument. Often it is easiest to approximate a real situation by successively 'less ideal' idealisations: eg in talking about chalk marks on a blackboard it is easier to develop number theory abstractly and then say that actually there are only 10^7 pieces of chalk in the world, than to try to develop chalk-numeral theory directly. Therefore, if any of our everyday reasoning is valid, so is idealisation in mathematics: there is no concrete-abstract gap to leap. It is a psychological fact that we do make a distinction between mathematical and non-mathematical aspects of computation-experiences; creating an autonomous discipline of mathematics, to focus on the necessary, 'structural' aspects of computations, is, at the very least, a natural step in systematising the way we reason.

But what do mathematical propositions *mean*, and are they literally *true*? To have a system of (apparent) propositions which are useful but not true would be unsatisfactory; it would suggest that we had not interpreted the propositions correctly. Perhaps by analysing their mode of usefulness we can attach a meaning to them in such a way that they are true (if they really are propositions) or (if they are not) are 'justified' in the appropriate sense.

Turning to the question '*what* is the idealisation process?', I will now describe it, without justification. We begin with our repertory of arguments, A, about the physical world, W. One statement of A is 'a repeatable operation can be repeated an extra time, if physical conditions permit' (for example, having written a chalk-numeral we can add a chalk stroke to form its successor, if we have not run out of chalk). Arguments in A need not use all the information available (eg, an argument about chalk-numerals need not consider the mineralogical properties of chalk); so we can remove some of the unused information (<u>abstraction</u>). The discussion of 'for any x, ... , regardless of x' statements in the previous section is an example of this, in which the value of x is 'removed information'. We may also drop physical-limitations qualifications specific to the 'aspects' we wish to discard (<u>idealisation</u>): in the example we delete everything after the comma, to get 'the (anonymous) operation can be repeated an extra time'; or, in other words, 'any number has a successor'. We do not add any new statements, such as 'the operation can be repeated ω times'; 'any number has a successor' already serves the purposes of what is called potential infinity. Call the idealised system of arguments A'.

It might be supposed that the arguments A' apply to an ideal world W' with, eg, an infinite supply of chalk. In fact, it is important to be clear that <u>idealisation is an operation on arguments: it transforms A to A', not W to W'</u>. A', at this stage, applies to W (with limited reliability), not to W'.

The next step is <u>reification</u>. We would like to replace idealised talk about objects by talk about ideal objects. For example, a symbol in an alphabet is the common type of certain tokens; a function is the common form of certain computation-experiences ('computations of the function for different arguments at different times'). Mathematical objects are 'universals' in the sense of general philosophy. If we understand sentences of the form 'ξ is a token of type "a"' or 'ξ is green' (in general, $P(\xi)$), we rephrase them as 'the symbol "a" is represented by ξ' or 'green is the colour of ξ' (in general, $\xi \in P$); then we treat "a" or greenness (in general, P) as an (abstract) object, that is, usable as a noun in any sentence and substitutable for a variable. We also say that it *exists*, or rather that there exists a unique x with all the required properties.

Justifying this is a serious philosophical problem. Perhaps all talk about abstract objects could be translated into talk about physical objects, so that there is no new ontological commitment; perhaps abstract objects do literally exist, as concepts or objects of thought; perhaps even so-called physical objects are abstractions from sense perception. I do not intend to take up a position on this question, for the following reasons.

(i) It is not a *mathematical* problem. I am attempting to reduce the problems of mathematical philosophy to special cases of general philosophical problems.

(ii) I study the foundations of mathematics because I seriously doubt the sense and truth of classical mathematics. I don't doubt the legitimacy of talking about greenness.

(iii) I suspect the problem is insoluble, for lack of any more basic notion by which to explicate universals.

In short, if I understood $\aleph_1$ as well as I understand greenness, I would consider my problems solved.

So I simply assume that it is legitimate to talk about 'the object P' (including to say that it exists) whenever sentences of the form $P(\xi)$ are meaningful (not necessarily as propositions). (Russell's paradox is avoided by being careful about meanings; many sentences $P(\xi)$ I do not regard as propositions, so it is unclear what would be a suitable notion of negation. The closest one can get to Russell's paradox is something like the *semantic* paradoxes which I will consider in §0.4.) Moreover, all mathematical objects are to be obtained in this way: it makes no sense from my point of view to have an abstract object which is not the result of any abstraction. We can picture the objects as inhabiting a world W', as long as in our *arguments* we use only A', not considerations suggested by treating W' as a physical universe. In particular, quantifying over W' clearly goes beyond what I have allowed.

Uncritical use of the W' picture would suggest that we could express number-theoretic propositions (eg Goldbach's conjecture) as objective statements about the infinite supply of pieces of chalk arranged in order. The problem would arise of justifying W''s 'existence' (as opposed to the existence of its inhabitants), as well as whether it is uniquely determined by A'. My procedure allows us to say 'every number has a successor' but not 'Goldbach's conjecture has a truth-value': in the usual terminology, potential infinity without actual infinity. There is no possibility of nonstandard models of A' because there *are no* models in the sense that would be required: when I say that I 'grasp the unique number system N', I simply mean that I know how to count. (For example, in protologic 'for all $n \in \mathbb{N}$, A(n)' (for decidable A) is represented as a relation between (essentially) the counting algorithm and A.)

Another problem is: can the 'structural' aspects of computation-experiences really be separated cleanly from other aspects, and are they logically necessary? To call a proposition necessary is to assert that no circumstances would count as a refutation of it. '*All black cats are cats*', for example, seems necessary because we cannot imagine an animal which we would be prepared to call a black cat but not a cat. This may just be the fault of our inadequate imaginations however: how can we predict how we will arrange our concepts in unforeseen future situations? Yet a non-feline 'black cat' would involve such an upheaval in our thinking that there is nothing we can intelligently say about such a beast now; so we exclude the possibility. Likewise in mathematics: we must rely on our most lucid and persuasive notions if we are to think at all, and if they are wrong or contain concealed confusions the best way of discovering this is to explore their consequences, while keeping an open mind about fundamentals.

Note the role of psychological terminology in mathematics: it is used to help define mathematical objects, but mathematical objects are not defined *as* psychological events; eg to define an algorithm one must explain how to execute it, but the algorithm is not any particular execution event.

I haven't defined the idealisation operation precisely: I merely indicated it with an example. The easiest way of specifying it is to describe A'. The arguments in A' concern successive operations: successiveness in general, in fact. Thus I need some basic assumptions about the geometry of time. Some of these will be purely contingent; the others are consequences of the idealisation.

__Assumption (α):__ Time is a partial order. The order relation is given by one mental event involving a memory of another. This has an idealised component (assume we never forget anything, which gives transitivity) and a contingent component (subjective free will, which gives antisymmetry).

__Assumption (β):__ Linearity. Time is a linear, or total, order. This is purely contingent.

__Assumption (γ):__ Time is discrete. A computation is completely specified if we state how to start it, what states are halting states, what result to deliver for a halting state, and how to continue after a non-halting state. In other words, ω (the first limit ordinal) is not finite (cannot be counted up to). This is also purely contingent.

__Assumption (δ):__ Homogeneity. A given computation produces the same result regardless of when it starts and how fast it proceeds. This is an unavoidable consequence of the idealisation approach. Granted the contingent fact that there are at least two events in time, this implies that every event has a later event at which we can continue the present computation. (For the given two events could be interpreted as a successful instance of counting from 0 up to 1; this implies that all such 'counting up to 1' processes succeed; hence for any mental event we can start a 'counting up to 1' process in parallel with whatever else we are doing, and we are guaranteed a successful outcome, ie a future event at which we reach 1.) There is a genuine philosophical problem of the existence of future events, and a physical question of whether we will survive long enough to see them: neither is any concern of mathematics.

Assumption (γ) restricts mathematics to recursive function theory. We can easily imagine it false. Suppose we resolve to test the computations A(0), A(1), A(2),... in succession to see if any one results in True. Suppose at some stage in the future we notice that we have tested A(0), and for each A(n) tested we have also tested A(n+1). Then by consulting our memory we can obtain a truth-value for $\exists n \in \mathbb{N}\ A(n)$. This gives an 'infinitary' version of mathematics in which quantifiers over $\mathbb{N}$ are directly meaningful because they can be evaluated by computation. With the means of expression so obtained we can define a universe of mathematical objects; quantification over the universe would still be illegitimate. I shall consider infinitary mathematics in Chapter 3.

Infinitary mathematics is reminiscent of ZF set theory in that both admit a broader notion of finiteness (ie numbers one can count up to, or 'limitedness') in which ω is finite. In my view, the reason ZF is not *immediately* inconsistent is that it is a distorted version of infinitary mathematics (cf §3.2). Quantifying over some such 'fixed' infinity is free of the problems discussed in §0.2: so it is not ruled out that the fixed infinity could be coherently regarded as a world external to ourselves, and that classical logic could be applied to it. But this does not go without saying. It requires an infinity assumption, that for a sequence of events E_0, E_1, E_2,... there is an event E_ω after all of them.

Whereas in finitary mathematics it is merely assumed that every event has an event following it. Finitary mathematics is developed in Chapters 1 and 2.

Assumption (β) can also be supposed false. In the many-worlds interpretation of quantum mechanics the world is continually branching into causally disconnected parallel worlds which could be considered as temporally incomparable. Of course this makes no difference to anyone's memory, and hence their perception of their experience, as long as every initial segment of time is linearly ordered. However, in exceptional circumstances it is possible to recombine the parallel worlds; thus one could hope to evaluate $\exists n \in \mathbb{N}\ A(n)$ by assigning one n to each parallel world and collating the results afterwards. In fact, it turns out that this can't be done in quantum mechanics: functions computable by quantum computers are computable by classical computers and vice versa. Nevertheless, this does illustrate that a denial of Assumption (β) is not absurd or inherently unacceptable.

Assumption (δ) has interesting consequences. Consider the question of the existence of a number, say 10^{100}: what this means is that we have an algorithm ('counting up to 10^{100}') and we want to know whether it will halt. Consider a computation-experience A in which we execute this algorithm. Now consider an alternative computation-experience B in which we execute the same algorithm but more slowly. The first step of B is like the first step of A; but in subsequent steps B adopts a delaying tactic, perhaps repeating the same step several times before admitting it as correct and passing to the next, so that B takes twice as long as A to do the second step, four times as long to do the third step, and so on.

Now by Assumption (δ) A must halt eventually iff B does; we cannot allow the existence of numbers to depend on how fast our brains are working, or how thoroughly we check our results, or when we stop for coffee breaks.

This may seem wildly unrealistic. Dropping the idealisation for a moment, the time disparity between A and B means that quite early in the computation either B is taking the lifetime of the universe to complete a step or A is taking less than 10^{-43}s (about the smallest physically meaningful interval of time, corresponding to the Planck length). One might argue that any foundational theory will have to justify existing mathematics, and hence eventually physics; and that in physics there is a clear concept of time measurement, in terms of which Assumption (δ) is absurd; and hence this suggests that mathematics should not be founded on Assumption (δ) after all. But this argument fails on its own terms. For in physics time measurements have no absolute significance but are merely an arbitrary assignment of numbers to events. It is true that in general relativity there is a well-defined invariant local 'proper' time. But this is just a conspiracy between geodesics; it tells us what the Lagrangian is doing in the vicinity (local weather conditions, so to speak).

Physics is constitutionally incapable of providing a more absolute measure of time than this. Two disjoint intervals of time are qualitatively different; and to call one longer than the other is meaningless without reference to some arbitrary measure, which we cannot allow if mathematics is to survive as an autonomous discipline focusing on the 'structural' aspects of computation-experiences. The idealisation has thrown away any criteria for comparing disjoint time intervals.

Returning to our processes A and B, introduce a third, C, as follows. To execute C we simulate B with our right hand and, in parallel, A with our left hand, except that the left hand does not stop at 10^{100} but continues as long as the right hand does.

Now, if the left hand of C reaches 10^{100} then surely A halts (why should it matter what the right hand of C is doing at the same time?). Then, as I said before, B must halt; so the right hand of C halts, by which time the left hand of C has successfully counted up to $1+2+4+\ldots+2^{10^{100}}-1$. Concentrating simply on C's left hand, then, if 10^{100} is a number so is $1+2+4+\ldots+2^{10^{100}}-1$. Clearly this argument can be generalised to justify any primitive recursion: so all primitive recursive number-theoretic functions are total.

What we have here is a method for proving general results about computations by comparing two computations step by step. Indeed, this is about all we can do with computations. They are specified in a local way: how to start, how to continue from an arbitrary step, how to stop, plus the abstract notion of iteration. Iteration is not reducible to any more primitive notion: we have no global view of computations, except by executing them individually. All we can do is obtain conditional results relating two computations by exploiting the fact that the bare concept of iteration in each is the same.

This enables us to prove general results: if computation A(x) succeeds then so does B(x), regardless of x (ie without inquiring into the value of x). This induction on the course of a computation should be included in the protologic of §0.2, since it is a single judgement which looks at the structure of A and B but not x: x is simply a schematic variable. Moreover, since this uses fully all our knowledge about computations, I would insist that this protologic exhausts all our raw materials for arguments about computations.

The above argument for induction and recursion will be controversial, and many people would reject it and prefer to talk about number theory without any such assumption (Yessenin-Volpin [20], Parikh [15]). But it seems to me that, without induction and recursion, number theory fragments into numeral theories, which become uninteresting. For there is then no reason to believe that the numeral systems (0,1,2,3...), (i,ii,iii,...), (I,II,III,...) and (I, I.II, I.II.III,...) are isomorphic; number is supposed to be what numeral systems have in common, but now they haven't anything in common. The sequence of marks I, II, III,... is not very edifying in itself; the only reason for studying it is the belief that the 'I' marks can stand for arbitrary processes. The basic assumption without which arithmetic is unintelligible is that when we talk about doing something repeatedly we can discuss the 'repeatedly' part separately from the 'something'; and this entails Assumption (δ) and my arguments about processes A, B and C.

Assumptions (α)-(δ) are not strictly a *part* of mathematics but an *input* to mathematics. If the contingent assumptions were false or the idealised assumptions were outrageously unrealistic (if, eg, we were drugged and confined in a strait-jacket so that we were unable to carry out any repetitive actions at all) then we would be prevented from doing mathematics under Assumptions (α)-(δ); mathematics wouldn't be falsified. If it seemed worthwhile we could label mathematics under (α)-(δ) 'Brouwerian mathematics' (exactly as one speaks of 'Euclidean geometry'), and develop a non-Brouwerian mathematics more applicable to our experience.

Thus mathematical propositions are analytic truths of the form: if the geometry of time is such-and-such then so-and-so follows. This is pure logic. I mean *informal* logic: not a particular predicate calculus system, but a general theory of (usually unspecified) objects, functions, relations, etc., applicable to all kinds of reasoning. There is no specifically mathematical primitive notion which has to be added to logic. The concept of iteration, for example, is already implicit in logic, since it is involved in characterising recursively the logical consequences of a set of axioms and inference rules: essentially this is because logical reasoning, like computation, has to take place *in time*. Thus my view of mathematics is logicist as well as formalist and intuitionist.

§0.4 Open-endedness

I have identified open-endedness as the main problem in formulating a foundation for mathematics: I will now explain how I propose to handle it.

Notice that under my characterisation of mathematical objects (§0.3) there can only be countably many of them, since they are all definable. Or rather, the mathematical objects are embedded in a countable infinity, since not all alleged definitions are 'valid' (ie actually define objects). The semantic paradoxes apply here, most importantly Berry's paradox ('the least number not definable in <100 characters') since it doesn't involve infinite quantification or any objects more exotic than numbers.

For finitary mathematics, 'validity' means simply 'seeing' that an algorithm always halts and so defines a total function, or that a tree is well-founded, or that a 'finitary' argument is convincing. Taking something like well-foundedness as primitive may seem odd: one cannot define it in ordinary language without saying 'for all branches,...', so it seems as if I am presupposing the meaningfulness of universal function quantifiers. But we are not bound by language usage, which simply reflects traditional ways of looking at things. I take well-foundedness as primitive, and explicate quantifiers in terms of it and other primitive notions. Of course, with any primitive notion we cannot define it in terms of anything logically prior; we can only give synonyms, suggest informal ways of looking at it, give examples, or indicate the role it is to play in elucidating other concepts.

The semantic paradoxes are usually considered resolved by distinguishing formal languages of different levels: each language is such that terms expressed in it are automatically valid, but there is no maximal language. But a formal language is just a meaningless system of symbols until we give it a meaning, which we do ultimately in an informal language such as mathematical English. When we call a formal language, L, *acceptable for use in mathematics* we mean that every informal definition which begins by describing L and then exhibits the *definiendum* as a term in L, is valid. Thus acceptability of languages is like validity of definitions except for the extra infinite quantification. The real paradox here is how anyone came to regard asserting infinitely many things valid at once as more secure than asserting them valid one at a time. Perhaps the idea was to choose a fixed language and do all mathematics inside it, so reducing the validity problem to a single assertion that the language is acceptable.

So as soon as a philosopher justifies the fixed language all of mathematics is justified. But the only reason for believing that the language ever can be justified is the underlying informal ideas about what we are entitled to define, which if they are reliable at all urge us not to stop at any fixed formal language.

The classical picture has a set of (coded) definitions and a universe V of objects: the denotation mapping from the former to the latter is called 'vague' or 'illegitimate' on account of the semantic paradoxes. This approach is not open to me, for I *obtain* objects *from* their definitions: the denotation mapping is perfectly clear, where it is defined (it takes a definition D to the object which involves 'executing' D). Questions can only arise, therefore, over its domain, ie validity. An accusation of vagueness is a handy device in foundational disputes because, for a fundamental concept which cannot be formally defined in terms of anything more primitive, it is so difficult to refute. Nevertheless it seems to me that we do have a clear ability to read alleged definitions and decide whether we are convinced that they have really 'constructed' or brought to our attention unique objects. (I include in the definition any accompanying explanation of why we should accept it as valid.) This seems a natural founding concept for mathematics, bearing in mind Criterion (α) of §0.1. At any rate, if I am deluded in believing that we have this ability we might as well give up doing mathematics, let alone justifying it, for nothing we ever think we have defined can be relied upon to make sense.

In finitary mathematics the validity problem arises in the halting of algorithms, ie what total functions exist. The intuitionistic meaning given to the quantifiers in §0.2 gives no clue to this question, for the meaning of $\forall x \exists y\ R(x,y)$ refers to the existence of a function producing a y for every x, and vice versa functionality can be expressed in $\forall\exists$ form.

We can construct an open-ended hierarchy of decidable classes of total recursive functions as follows. Primitive recursive functions are total (by §0.3); therefore the universal function over primitive recursive functions, ie F defined by

$F(m,n) \equiv$ (m'th primitive recursive function)(n),

is total (and by coding pairs we can turn it into a function of one variable). So everything primitive recursive in F is total. So the universal function F' over the functions primitive recursive in F is total. Moreover, the operation which obtained F' from F always takes total functions to total functions. So we can iterate it to get F'', F''',..., and together they give the 'union' F^{ω}: $(n,m) \mapsto F'''^{\cdots '''}(m)$ (n primes). In fact, given any operation taking total functions to total functions (eg $F \mapsto F'$) the result of iterating it in this way gives a new operation ($F \mapsto F^{\omega}$): this gives us a mapping from operations to operations. And we can continue in this way, defining ever higher-type operations, all valid in the sense that they ultimately lead to total recursive functions, at each stage generalising something we did earlier. There is no point trying to replace this kind of argument with a formal system for proving functions total because we could always transcend such a system by constructing the universal function over the provably total functions, which would be obviously total but not provably so. We must decide to leave totality arguments unformalised if we are not to obscure the open-endedness.

A similar phenomenon occurs in infinitary mathematics. Take for example 'ω-infinitism': that is, assume we may evaluate as computations quantifiers over $\mathbb{N}$ but no higher infinity. Then we can define arithmetic subsets of the natural numbers; because these can be enumerated we can define a universal function over them, whence a new non-arithmetic (but still valid) set of numbers; and so on as before. Validity of an alleged definition of a subset of $\mathbb{N}$ will then involve the well-foundedness of the tree implicit in its definition, which will not be expressible in ω-infinitism. In general, the validity question cannot be removed by any infinity assumptions. However I shall stick to finitary mathematics here for definiteness.

One can, of course, 'prove' a definition valid by verbal arguments in English, as I have just been doing; the argument can then be included as part of the definition so as to make it manifestly valid. But the problem of validating informal arguments is essentially the same as the problem of validating objects. The argument merely 'displays' the way in which the object is arrived at: it needs an inexpressible act of 'intuition' to to see that an argument, which is merely a string of alphabetic characters, makes sense (or rather, one *can* express the act of intuition, but then the expression itself has to be validated: so one hasn't really expressed *why* it is valid).

Clearly, the levels of recursive functions constructed above can be indexed by constructive ordinals (though the operations cannot): validity may be regarded as a question of whether ordinal notations are well-founded (ie really denote ordinals). Thus validity can be thought of in several ways:

(t) totality of functions;

(*ii*) convincingness of 'finitary' arguments;

(*iii*) well-foundedness of ordinal notations (cf the familiar idea of measuring the strength of a formal system by a proof-theoretic ordinal);

– and possibly other ways.

What is fundamental here is our ability to formalise our past informal arguments. Suppose at a certain time we have a formal system F and our repertory I of informal arguments for talking about it. At a later time we decide that, because this is mathematics and not mysticism, we should be able to formalise the arguments we were using earlier. We cannot formalise all of I, but we don't need to; for we see that earlier we only used a decidable subclass I_0 of I. In fact, we only used finitely many arguments from I, but we can now see that the reason they convinced us at the time was because they were all of a certain form I_0: thus in picking out I_0 we are generalising our past arguments. We then build a new formal system $F + I_0$. This transcendence step from F to $F + I_0$ is the key to all that is problematic in the foundations of mathematics; it is illustrated by the above arguments constructing levels of total recursive functions. Formal systems can never replace informal argument, but only particular classes of informal arguments.

The fact that we cannot formalise all our arguments means that it is not in general legitimate to use the concept of validity in our arguments; for an argument or definition A involving validity will depend for its meaning on the meaning of validity, which in turn depends on the meaning of A; so that A is circularly defined. Notwithstanding this, some arguments and definitions using validity are valid; for example, if 'validity' only occurs in phrases of the form 'valid for α'

(where $\alpha \in$ a well-ordered set) and 'valid for α' is defined in terms of 'valid for β',...'valid for γ' where $\beta,\ldots\gamma<\alpha$ (an example of this is validity for a formula, §2.1). An example of where invalid use of validity leads to a contradiction is the 'definition':

$\mathfrak{F}(n,m) \equiv$ (n'th valid (ie total) function)(m),

where possible function definitions are enumerated lexicographically; if $\mathfrak{F}$ is validly defined then $(\lambda n.\ \mathfrak{F}(n,n)+1)$ is also valid, whence a contradiction by the usual diagonal argument.

In view of the unaxiomatisability of mathematics it may seem pointless to give any further global account, since all such an account would do would be to split off a formal part F from our informal arguments I, leaving the remaining part I\F essentially just as complex as I. However, recalling Criterion (γ) of §0.1, it is worth explaining systematically how mathematics appears to refer to actual infinities in quantifiers, ie to formalise predicate calculus and show how it justifies arithmetic and analysis. The formal part F will consist of the protological sequent calculus (§1.5) plus the definition of the proof predicate (§§2.0 and 3.4). The leftover part I\F will from now on be what is meant by 'validity': it will consist of seeing informally that a tree is well-founded, or that a function maps well-founded trees to well-founded trees, or some such concept. Particular formal systems for arithmetic and analysis will be justified by the formal part and finitely many validity judgements (cf the usual axiomatic approach of splitting mathematics into a formal part (derivations in an axiom system) plus finitely many judgements that the axioms and rules are sound: I cannot proceed like this because I do not regard quantified formulae as genuine propositions).

I may have given the impression that human minds have the intuitive ability to 'understand' arguments and definitions in a way that transcends recursive or 'mechanistic' description: this does not follow at all. Whether we are all subject to Church's Thesis depends on the fundamental laws of physics and the architecture of the brain; but suppose we are. Then a particular mathematician will recognise a decidable class C of definitions as valid. If everything in C is valid then the universal function U_C over C is also valid, but U_C is not in C so he will not recognise U_C as valid. Even though he may understand the preceding argument, and appreciate that it applies to himself, he cannot construct U_C because he cannot know C, because he cannot know his own code number (as a Turing machine). This is the meaning of the above argument that the function $\mathfrak{F}$ is invalid: the definition of $\mathfrak{F}$ assumes that we can simulate our own activity, which no Turing machine can (but $\mathfrak{F}$ is invalid regardless of Church's Thesis). The same result also follows from Gödel's incompleteness theorem, for if we knew our own code numbers we could construct our own Gödel sentences. Another mathematician with a larger C could recognise the first mathematician's U_C as valid but not his own U_C.

Thus there is no reason to posit a non-recursive class of objectively valid definitions. Every mathematician has his own horizon which he himself cannot see, although he can see everything up to it. This is a relativistic picture, in which mathematics looks different from different viewpoints.

By contrast, ZF set theory is curiously reminiscent of medieval cosmology, with its static, finely graded hierarchy, which is divided into three 'realms' as follows. There is the 'mundane' realm (the first

$\omega+\omega$ levels of the cumulative hierarchy, say), which is the proper area of human concern; then comes an 'astronomical' realm (large cardinals) populated by objects quite unlike anything at the mundane level, which we can speculate about forever without hope of certain knowledge. Axioms of infinity are piled on top of one another in an attempt to accommodate the intuition of open-endedness which is as obviously doomed as the attempt to describe an elliptical orbit by ever more epicycles within epicycles. (The preoccupation with formal consistency at the expense of truth in intended models and clear informal ideas is much like the pre-Keplerian astronomical concern with 'saving the appearances' at the expense of physical plausibility.) Finally there is the 'transcendental' or 'theological' realm (proper classes) of things which it is not even proper for us to discuss. Mathematics seems ready for a more dynamic and observer-centred world-view; and it is this that I am aiming to provide.

Chapter 1: Protologic

§1.0 The naive notion of intuitionistic logic

In §0.2 I introduced the term 'protologic' for the part of quantificational logic I consider directly justifiable, and suggested that general predicate calculus formulae could be interpreted as incomplete statements based on protologic. In this chapter I shall show how to do this, this time starting from what is necessary for intuitionistic logic and working backwards to arrive at what protologic must be (it will turn out that it must be what I said it was in §0.2). For simplicity, I shall discuss it first in the context of *finitary* mathematics; infinitistic mathematics will be considered in Chapter 3.

Finitary mathematics is mathematics without any assumptions of 'actual infinity' (ie subject to Assumption (γ) of §0.3). The basic objects are natural numbers, and all other mathematical objects may be regarded as recursive functions on natural numbers. There is no hierarchy of 'higher type' functionals built on top of the recursive functions; there are no special variables for 'constructions'. The universe of numbers and recursive functions is complete in itself. All objects are codable as numbers because they are either numbers themselves or essentially algorithms (which should be thought of syntactically as programs). The 'open-endedness' of mathematics is expressed by the fact that the concept of *total* recursive function cannot be fully incorporated in the formal system; for given any recursive enumeration of a class of total recursive functions we can diagonalise out of it. Proving totality depends essentially on well-foundedness (see §1.4), which is a part of the general notion of 'validity', which is outside the formal system (as explained in §0.4).

The philosophical considerations of §0.2 compel us to restrict ourselves to intuitionistic quantification when talking about the whole universe, which in the case of finitary mathematics means quantifying over the natural numbers. So we need an account of intuitionistic logic.

We shall be interested in first-order formulae of arithmetic. Intuitionism sees the meaning of such formulae as determined by what 'constructions' count as proofs of them. This is defined by induction on the structure of the formula; the naive definitions, implicit in the verbal explanations of Brouwer and Heyting, are as follows, where '$P \vdash F$' means 'construction P proves formula F'.

(α) $P \vdash A$ iff A, for atomic A (ie A is a term which can be evaluated directly, so needs no extra proof); some people would insist that $P = 0$, or perhaps that P be a computation sequence for A, but this makes no difference.

(β) $P \vdash A \wedge B$ iff P is a pair (Q,R), where $Q \vdash A$ and $R \vdash B$.

(γ) $P \vdash A \vee B$ iff P is a pair (i,Q), where ($i=0$ and $Q \vdash A$) or ($i=1$ and $Q \vdash B$); thus P contains a proof either of A or B, plus an indication of which is proved.

(δ) $P \vdash \exists x\, A(x)$ iff P is a pair (n,Q) and $Q \vdash A(n)$; thus a proof of an existential formula is an instance plus a proof that the instance works.

(ε) $P \vdash \forall x\, A(x)$ iff P is a function such that, for all n, $Pn \vdash A(n)$: ie a proof of a universal formula is a general method for obtaining a proof of an arbitrary instance.

(ζ) $P \vdash A \supset B$ iff P is a function such that, whenever $Q \vdash A$, $PQ \vdash B$: thus a proof of a conditional is a general method for transforming a proof of A to a proof of B.

(It is usually convenient to define $\neg A$ as $A \supset$ False, so negation does not need a separate clause.)

Thus the logical constants are not truth-functional but *proof-functional*; for example, 'being a proof of $A \supset B$' is a function of 'being a proof of A' and 'being a proof of B'.

§1.1 What do formulae mean?

I have said that the meaning of a formula is determined by the class of its proofs. Is this a *definition* of its meaning or merely a true fact about it? To talk about 'proof' at all seems to presuppose a prior notion of 'truth' for formulae (otherwise what is the proof *proving*?), in which case surely the meaning of a formula lies in its truth-conditions, and the equivalence of truth to provability needs to be demonstrated meta-mathematically.

In fact, intuitionists generally regard a formula as somehow referring to the act of finding a certain construction (which is *then* labelled a 'proof'), or the demand for such a construction, or the assertion that we have already found one. The word 'proof' may be misleading: we could just as well read '$P \vdash A$' as 'P *realises, justifies, exemplifies, satisfies the test*, or *solves the problem* A'. But 'proves' is the best word; for, whatever we call it, it is intended to *replace* the classical notion of proving a formula, and to use a different word might suggest that a separate notion of proof survives from which $\vdash$ must be distinguished.

Thus in 'P $\vdash$ A' the part '$\vdash$ A' should be regarded as a single symbol: it is a test for constructions which is applied to P. A has no meaning beyond the meaning of '$\vdash$ A'.

This is all very well, but what *literally* does a formula *say*? Intuitionists often talk as if the formula A is logically equivalent to 'I have found a construction P such that P $\vdash$ A'. If this is so then formulae are contingent psychological propositions about the contents of particular mathematicians' minds which are initially false and become true when P is found. This conflicts with all we have ever assumed about the nature of mathematical statements.

Take for example the formula $\exists n\ A(n)$ where A is decidable. Suppose we discover by computation that A(5) holds. Then we usually conclude that $\exists n\ A(n)$ is 'true'. What exactly do we mean? We have two unpalatable alternatives:

(i) We can say that we have *discovered* that $\exists n\ A(n)$ is true, and that it always was true, always will be true, and would still be true even if no one had thought of trying n=5, even if no mathematicians had ever evolved, even if the universe had never existed.

If $\exists n\ A(n)$ had a truth-value even before we discovered it, that suggests that other formulae might have truth-values even if we do not know them and cannot determine them. This evokes a picture of a Platonistic 'actually infinite' universe about which number-theoretic formulae are objectively either true or false. But I argue in Chapter 0 that this picture is misleading: we cannot just insert an infinite quantifier in front of a proposition and expect the result to be a meaningful proposition. We can only give quantified formulae an 'indirect' meaning.

(ii) The second option is to say that $\exists n\ A(n)$ *was* false (or undefined) and *became* true when we verified A(5). Many intuitionists would accept this view: it is elaborated in the 'theory of the creative subject'. But we still need 'meta-quantifiers' to say what the creative subject can or will ever prove. Moreover, mathematics from this viewpoint threatens to lose its normative character and degenerate into descriptive psychology. It is unclear whether two mathematicians can ever contradict each other since they are each only reporting on their own mental experiences. At any rate, mathematics has been thoroughly mixed up with empirical considerations, contrary to Criterion (β) (§0.1).

The way out of the dilemma is to deny that formulae mean propositions at all. I prefer to think of a formula as an *exclamation* rather than an assertion; for example, $\exists n\ A(n)$ means 'An n for which A(n) holds!'. It does not assert anything, neither is it meaningless: its meaning consists in the fact that it is appropriate to utter it in some circumstances (having such an n) and not in others. It is incorrect to translate it as 'I have found an n for which A(n) holds': the two sentences are inequivalent. (This may be clearer if we consider questions instead of exclamations: 'Is it raining?' is not equivalent to 'I want to know whether it is raining', because the former is a query about the weather and the latter is a report on my state of mind.)

The point of making such an exclamation is that if ever in future we want an n for which A(n) holds we simply recall whether we have ever announced '$\exists n\ A(n)$!'; a proof of $(\exists n\ A(n)) \supset B$ is a construction which assumes such an event.

This interpretation of formulae is analogous to Kleene's view that they are 'incomplete communications' [9,§1] (see also my discussion in §0.2) and Kolmogorov's view that they are statements of problems [10]. In all cases the meaning of a formula consists in the truth-values it acquires when additional information (the 'circumstances', 'solution' or 'proof') is added. $\exists n\ A(n)$ has no truth-value of its own even though $A(5)$ is known to be true.

§1.2 Difficulties with the naive definitions

Having clarified what (α)-(ζ) are supposed to be defining, there are still difficulties with the definitions themselves. In the $\forall$ and $\supset$ clauses I said that a proof was a function of a certain kind. But how do we know, given just the function, that it has the property stated in the definition? The $\vdash$ relation should be decidable in some sense: if we cannot decide whether $P \vdash A$ for particular P and A then we have not been convinced of anything, so P cannot be said to have 'proved' or 'justified' A, so in fact P has *not* proved A.

Thus, many people would say that a proof of $\forall n\ A(n)$ or $A \supset B$ must consist of a function together with some sort of 'justification' or 'evidence' that the function works. The $\forall$ and $\supset$ clauses become:

(ϵ') $(E,f) \vdash \forall n\ A(n)$ iff E is evidence that, for all n, $fn \vdash A(n)$;

(ζ') $(E,f) \vdash A \supset B$ iff E is evidence that,

for all Q, $Q \vdash A$ implies $fQ \vdash B$.

This notion of 'evidence' needs to be able to justify statements of the form

$$(x)\ D(x) \qquad \text{and} \qquad (x)\ D_1(x) \rightarrow D_2(x),$$

or, without real gain in generality,

$(x,y,\ldots z)\ D_1(x,y,\ldots z),\ D_2(x,y,\ldots z),\ldots D_k(x,y,\ldots z) \longrightarrow D_{k+1}(x,y,\ldots z),$

where D, $D_1,\ldots D_{k+1}$ are decidable. The quantification $(x,y,\ldots z)$ includes the whole expression in its scope. I am writing (x) instead of $\forall x$, $\longrightarrow$ instead of $\supset$, and a comma instead of $\wedge$, to suggest that these simple statements (call them <u>sequents</u>) are different from predicate calculus formulae. The 'evidence' predicate cannot be simply $\vdash$, since $\vdash$ is defined by recursion on the structure of the formula, and the recursion step presupposes that we know how to justify sequents. The sequents themselves cannot be further reduced by the $\vdash$ definition so we will need a new notion of justification for them, which will probably use recursion on the structure of E (so that E is some sort of derivation tree). Recursion on a proof and recursion on the formula proved are both legitimate means of defining a notion of proof; but we must be clear which we are using at any stage to avoid circularity. This necessitates a rigid distinction between proofs and formulae (on one hand) and evidence and sequents (on the other).

We need to argue that sequents are more fundamental than predicate calculus formulae, and develop a 'protologic' for giving them a direct meaning, so that we can then use (α)-(δ),(ϵ'),(ζ') to give formulae an indirect meaning as exclamations, incomplete communications or problems.

In the next section we shall review several accounts of intuitionistic logic roughly along these lines.

§1.3 Systems of intuitionistic logic

The first attempt to formalise the idea of 'construction' and the role it plays in intuitionistic proof was Kleene's 1945 paper [9] on 'realisability'. Kleene defines a relation 'n r. A' ('n realises A', ie 'n proves A') essentially as in clauses (α)-(ζ) above, except that he insists that the functions mentioned be recursive functions coded as natural numbers. Thus his $\forall$ and $\supset$ clauses are

n r. $\forall x\, A(x)$ iff, for all x, $\{n\}(x)$ is defined and realises $A(x)$;

n r. $A \supset B$ iff, for all m such that m r. A, $\{n\}(m)$ is defined and r. B.

Kleene points out the lack of any notion of 'evidence' for justifying the claim that n does realise the formula. Thus his definition is 'not to be regarded as more than a partial analysis of the intuitionistic meaning of the statements' (§2).

Kleene considers that realisability makes explicit 'certain necessary and intuitionistically sufficient conditions that a proposition hold from the standpoint of the intuitionists' (§13): this statement, of course, is made from a classical standpoint.

Later versions of realisability have had even less to do with analysing intuitionistic logic and have been of purely proof-theoretic interest.

Also relevant is Gödel's 'Dialectica' interpretation of intuitionistic arithmetic in finite type theory [3]. Gödel defines a transformation from arithmetic formulae to Σ_2-formulae. The connection with realisability is that if A transforms to $\exists \underline{x} \forall \underline{y}\, A^{*}(\underline{x},\underline{y})$ (for lists of variables $\underline{x},\underline{y}$) we can regard $\forall \underline{y}\, A^{*}(\underline{x},\underline{y})$ as $\underline{x}$ r. A, and then A transforms to the statement that A is realisable.

Gödel shows that if A is derivable in Heyting arithmetic there are $\underline{\phi}_0$ such that $A^{*}(\underline{\phi}_0,\underline{\psi})$ (with free variables $\underline{\psi}$) is provable in finite type theory. Thus, the 'proof' of A may be regarded as Φ_0 together with the derivation of $A^*(\Phi_0,\underline{\psi})$. Here the protologic is free variable argument in finite type theory.

The drawback with this as an account of intuitionistic proof is that the transformation Gödel defines is weaker than the intended meaning of the logical constants. Gödel's $\supset$ clause is

$A \supset B$ transforms to $\exists \underline{V}\underline{Z}\forall \underline{y}\underline{w}\ [A^*(\underline{y},\underline{Z}(\underline{y},\underline{w})) \supset B^*(\underline{V}(\underline{y}),\underline{w})]$,

where A transforms to $\exists \underline{y}\forall \underline{z}\ A^*(\underline{y},\underline{z})$ and B transforms to $\exists \underline{v}\forall \underline{w}\ B^*(\underline{v},\underline{w})$ (ignoring any free variables in A and B). The informal motivation presumably behind this is that to prove $A \supset B$ we must transform an arbitrary proof $\underline{y}$ of A to a proof $\underline{V}(\underline{y})$ of B, and to justify this we must show

$$(\forall \underline{z}\ A^*(\underline{y},\underline{z})) \supset (\forall \underline{w}\ B^*(\underline{V}(\underline{y}),\underline{w}));$$

for which it suffices to derive $B^*(\underline{V}(\underline{y}),\underline{w})$, for an arbitrary $\underline{w}$, from a single instance $A^*(\underline{y},\underline{Z}(\underline{y},\underline{w}))$ of the antecedent.

This diverges from the full meaning of intuitionistic implication in that the latter allows us to obtain $B^*(\underline{V}(\underline{y}),\underline{w})$ from several (even infinitely many) instances of $\forall \underline{z}\ A^*(\underline{y},\underline{z})$ or from the presence of a general method of proving $A^*(\underline{y},\underline{z})$ for arbitrary $\underline{z}$.

Bishop [1] gives a somewhat similar argument; he advocates replacing the intuitionistic $\supset$ by Gödel's weaker version since the latter is easier to work with. However, I am trying to understand the full intuitionistic notion as it is.

Of course, Gödel does not claim that his interpretation is an explication of the intuitionistic logical constants: it is a proof-theoretic device. But surely its success as such is due to its rough correspondence with the intended meaning. It suggests that intuitionistic logic *can* be described in this way, and we do need a separate notion of protologic, but the implication clause will have to be broadened.

Kreisel has developed theories intended to do just this, [11], [12]. He assumes we have a decidable notion $\pi(P, \forall x\ \mathfrak{v}(x))$, meaning P is a proof (in the sense of protologic) of $\forall x\ \mathfrak{v}(x)$. Here P is a 'mathematical object' and $\mathfrak{v}$ is a 'notion' (ie a decidable property of mathematical objects). For Kreisel, a 'mathematical object' may be something 'abstract' (such as a function, construction or proof) rather than 'concrete' (a 'spatio-temporal configuration'). If I understand him correctly, to say that a proof is not a spatio-temporal configuration means that although we can represent it syntactically we cannot represent syntactically what distinguishes valid proofs from invalid ones; whereas a proof as an abstract object is manifestly valid. Similarly a function can be represented as a concrete algorithm, but because the halting problem is unsolvable we cannot express why it is totally-defined: to recognise it as total we have to 'grasp' the corresponding abstract object.

Thus Kreisel imagines a stratified universe with concrete objects at the lowest level, functions and constructions over them at the next level, functions and constructions over *them* at the following level, and so on.

Kreisel does not define his π predicate but gives alternative systems of axioms and rules for it, which constitute his protologic. He assumes that for every formal protological proof P of $\forall x\ \mathfrak{v}(x)$ there is an abstract proof α_P such that $\pi(\alpha_P, \forall x\ \mathfrak{v}(x))$. One version assumes a reflection principle

$$\pi(P, \forall x\ \mathfrak{v}(x)) \longrightarrow \mathfrak{v}(t) \qquad (t \text{ is a term}).$$

He then suggests various ways of obtaining logic from protologic; in the simplest version the $\supset$ clause is

$$(Q,T) \vdash A \supset B \quad \text{iff} \quad \pi(Q, \forall x\ x \vdash A \Rightarrow Tx \vdash B),$$

where $\Rightarrow$ is a truth-function.

N.D.Goodman, [4], [5] and [6], develops these ideas further. In [4] he uses a type-free universe of partial constructive functions (possibly the partial recursive functions); in Kreisel's terms all his objects are spatio-temporal configurations. This brings him much closer to my viewpoint.

Goodman's $\supset$ clause reads

$(y_1, y_2) \vdash_x A \supset B$ iff y_1 proves: for all z, if $z \vdash_x A$ then $y_2z \vdash_x B$,

where x is an assignment for the free variables of A and B. He argues that $\vdash$ must be decidable, so that truth functions will work classically on it. This allows him to define (if...then...) as a recursive function, ie something simpler than $\supset$, so the definition of $\supset$ escapes circularity. In fact, (if $z \vdash_x A$ then $y_2z \vdash_x B$) evaluates $z \vdash_x A$ first, and only if the result is 'True' goes on to evaluate $y_2z \vdash_x B$: this is a precaution in case y_2z is undefined for some non-proofs z.

If $\vdash$ is to be decidable we must make the protological proof predicate (for formulae $\forall z\ u(z)$=True) decidable. But Goodman shows that this leads to a contradiction by a diagonalisation argument. So he recovers by stratifying the universe of constructions according to their subject-matter: 'proofs must be *about* objects already constructed', he insists [4,§10]. The new protological proof predicate has an extra argument v, which is a 'grasped domain' (level of the hierarchy): it is now decidable whether w is a proof of $\forall z \in v\ u(z)=\mathrm{True}$.

Goodman finds it necessary to introduce a reducibility operator F: for each 'rule' z and domain a, $Faz \in a$ and Faz extends z on $a \times a$. Thus Faz is a 'representative' of z at level a. Then in proving $A \supset B$ we need only consider proofs of A below a certain level since higher level proofs are already 'represented' lower down. Thus the impredicativity in the definition of $\supset$ is shifted to the operator F. Goodman says [4,§11] 'It seems to us essential to the intuitionistic position that given a fixed assertion A about a well-defined domain, there is always an *a priori* upper bound to the complexity of possible proofs of A.'

It doesn't seem at all essential to me. I believe constructive arguments but don't believe in Goodman's reducibility axiom, even after it has been explained to me how the former supposedly depend on the latter; thus reducibility does not seem to be part of my notion of constructivity, at least. Weinstein [19] agrees with me.

In any case, notice that F and the protological proof predicate don't occur at any level of the hierarchy. This is what is so disconcerting about stratified theories generally. We are told that, on general grounds, every well-defined object must have a 'level' or 'type'; then we are shown some that don't. We are left with the impression that 'well-defined object' must have been meant in a narrower sense than we thought, but what sense is never explained. And what is to stop us from extending the universe by levels of things definable in terms of the 'type-free' objects?

In Goodman's theory (which is equivalent to Heyting arithmetic) the only grasped domains are the 'basic' domain B and all domains obtainable from B by finitely iterating the operation E: EX includes X, proofs about X, and certain other constructions. We can 'grasp' $\bigcup_n E^n(B)$, which we might call $E^\omega(B)$, in the informal sense but not in Goodman's technical sense. This indicates that the technical sense is inadequate: it does not encompass the open-endedness of our constructive abilities.

In [5,§6] he restates his position in explicitly Kreiselian terms, argues unconvincingly for reducibility, and explains why he won't allow $E^\omega(B)$: the obstacle is that 'The rule which leads from the n'th level to the (n+1)'st level is not a rule which we can understand'. Yet in the previous paragraph he explained his hierarchy of levels by introducing levels 0, 1, 2, and saying, 'Continuing in this way, we can construct the n'th level for arbitrary n'. Continuing in *what* way? Why explain something that cannot be understood? His truncation of the hierarchy at level ω is surely a refusal to face squarely the open-endedness issue.

Of course, the whole point about open-endedness is that we can transcend any given formal system. But it seems to me (§0.4) that a theory of constructions (ie mathematical objects) should accommodate the open-endedness by leaving a corner of the system undecidable (what I call 'validity').

A more recent theory of constructions is Martin-Löf's 'type theory' [14]. This uses the "proposition = type" idea. A proposition in intuitionism is identified with the class of its proofs; this class is then called a 'type'. The basic formula of the theory is $a:A$, which may be read as "a is an object of type A" or "a is a proof of the proposition A".

'A proof of $(\forall x \in A)B(x)$ is a function which to an arbitrary object of type A assigns a proof of $B(x)$.' In other words, $(\forall x \in A)B(x)$ is simply the product type $(\Pi x \in A)B(x)$. Similarly, 'a proof of $A \supset B$ is a function which takes an arbitrary proof of A into a proof of B'. Thus $A \supset B$ is the function type $A \to B$ (ie the type of functions from A to B). An existential proposition is a sum type, and so on.

Using the word 'type' in this way involves combining two distinct ideas:

(i) the intuitionistic notion that a proposition is defined by reference to the class of its proofs,

(ii) the classical doctrine of type theory that every object must have a type because it is illegitimate to talk about the whole mathematical universe at once.

I cannot see what these ideas have in common to justify this usage. Interpreting universal statements as product types, implications as function types, etc., is technically elegant but seems to ignore the need for protologic altogether. I say 'seems to' because protologic is actually present in Martin-Löf's system, though it is somewhat obscured. Martin-Löf considers that propositions of the form $a{:}A$ are decidable, because every object carries its type with it so that we merely need to compare a's type with A. Thus, for example, there is no object '2': but we have 'the natural number 2' and 'the even number 2'. There is of course a mapping f taking even numbers to the corresponding natural numbers. This f is not, as one might hope, a proof that all even numbers are natural numbers, but merely that if there is an even number then there is a natural number. Indeed, it is difficult to see how an even number can possibly be a natural number.

Maybe I am misunderstanding this, or maybe this is all as it should be. At any rate, this bears directly on what the concept of proof means. For a proof of $A \supset B$ it is not enough to define a function f which happens to take every object of type A to an object of type B; f must be constructed *as* a function of type $A \to B$.

'Constructing a function' usually means defining it in such a way that it can manifestly be calculated: it needs no existence proof to justify it. Here we must 'construct' f in such a way that it manifestly maps A into B. Martin-Löf gives a formal system in which this can be done: this is effectively his protologic. With Sundholm [16], we should distinguish the 'process of construction' from the 'object constructed'. The 'process' (the proof of $f{:}A \to B$) is self-justifying because we can examine each step and accept it as correct; the resulting 'object' (f together with its type $A \to B$) is not because it is not obvious from the form of f that its type is $A \to B$. But this is just another way of saying that the function alone is insufficient as a proof: we need a justification in some protological system simpler than, and prior to, predicate logic.

Martin-Löf's system is, like Goodman's, highly stratified. Objects belong to types which belong to 'universes' V_0, V_1, V_2,... (which satisfy reflection principles). $V_0 \in V_1 \in V_2 \in \ldots$, and a function which takes each number i to an object in V_i cannot belong to any V_n, so presumably doesn't 'exist'. Thus my remarks about Goodman's stratified system apply equally to this one.

§1.4 Informal account of protologic

My reasons for dissatisfaction with the systems of the previous section are

(i) They don't take proper account of the essential open-endedness of the constructive universe and our valid modes of proof. One cannot of course formalise this notion, but one can create a space for it within the formal system. Goodman and Martin-Löf have stratified systems which stop at an arbitrary level. My approach (§0.4) is to have an unstratified universe of 'alleged constructions', only some of which are 'valid', validity itself being unformalised.

(ii) They don't explain how the protologic arises from our basic experience of computation (or whatever finitary mathematics is supposed to be based on) and how it is philosophically prior to predicate logic.

(iii) They don't define the universe of constructions or the protologic, but simply state axioms and rules about them (except for Kleene, who has the universe equal to the natural numbers and no protologic at all). We are not here in the position of a classical set theorist speculating about an inaccessible set universe and looking for statements which 'force themselves upon us as being true'; we are supposed to be talking about our own constructions, with which we should be more familiar. Thus the axiomatic/model-theoretic approach of classical foundational mathematics is inappropriate. Martin-Löf actually produces a model of his system, the model of 'closed normal terms', without saying whether it is the intended model. It seems to me he ought to know: presumably the closed normal terms are names denoting objects which must be in the intended model, but would he ever accept a proof that wasn't a closed normal term?

It seems clear that we do need a protologic, to handle 'sequents' (in the sense of §1.2); it must be foundationally prior to predicate logic and not use the usual intuitionistic definitions (α)-(ζ) of §1.0.

From the arguments of Chapter 0, the basic propositions of mathematics (roughly analogous to experimental data in science) are assertions that a certain computation will eventually 'succeed', ie halt with a certain kind of result (the result 'True', say, without loss of generality). Thus the most general kind of statement we can make is of the following 'general-hypothetical' form

$$(x)\ A(x),\ \ldots\ B(x) \rightarrow C(x),$$

meaning that whenever we have had the experience of successful computations of the form A(x),...B(x), for arbitrary x, then the statement predicts that C(x) will also succeed. This is refutable in that C(x) may halt with a non-True result, but even if we try to compute C(x) for a long time without the computation halting the statement is still meaningful in that it asserts that it is worthwhile to continue. By contrast, a more complex statement such as

$$(x)\ [(y)\ A(x,y) \rightarrow B(x,y)] \rightarrow C(x)$$

has no such clear predictive meaning.

Clearly we may replace x with several variables. This gives exactly the sequent form required for protologic (§1.2).

(x) A(x),...B(x) $\rightarrow$ C(x) should be read as a single statement 'A-ness,...B-ness entail C-ness' rather than an infinite conjunction of particular statements. It really asserts that computations of C are 'connected' with computations of A,...B in such a way that success in A,...B entails success in C. This is an incomplete statement in that it does not say what the 'connection' is: that information will be implicit in the argument we use to derive the sequent.

The major tool we have for 'connecting' computations is induction on the process of computation. Computations A and B are connected if we can relate the 'starting position' of A to that of B, the 'halting condition' of A to that of B, the 'next step' operation of A (which tells us how to continue from an arbitrary non-halting state) to that of B, and the final 'result extraction' part of A to that of B. Other similar kinds of connection are possible, all deriving from my idealised assumptions about computations in §0.3.

This is obviously an induction argument, yet I have not mentioned numbers. Number induction is simply a special case of this in which one computation is a counting process. In fact it is not necessary to define a special sequence of objects 0, 1, 2,...; any sequence generated from an object 'a' and a function 'S' guaranteed to produce new objects each time it is applied will do. Therefore induction should be embodied in protologic in the above number-free form: I will then introduce objects called '0', '1', '2',... and prove the numeric principle of induction in terms of them.

Induction allows us to derive sequents from other sequents, so the sequent we are trying to obtain will appear as the root of a derivation tree of sequents. However, induction only allows us to obtain sequents with variables x,...y from other sequents with the same variables. The ultimate source of quantified assertions must be 'schematic' (free-variable) arguments. This is where a sequent $(x,\underline{y})\ A(x,\underline{y}),\ \ldots\ B(x,\underline{y}) \longrightarrow C(x,\underline{y})$ (where $\underline{y}$ is a list of variables) is derived by treating the sequent as a schema and x as denoting an unspecified object, and deriving $(\underline{y})\ A(x,\underline{y}),\ \ldots\ B(x,\underline{y}) \longrightarrow C(x,\underline{y})$ 'regardless of x', ie without reference to the value of x. As explained in §0.2, this is the sole source of our ability to make general statements.

Since protologic is intended to deal with the success of computations relative to other computations it is similar to the 'calculi for program correctness' developed by theoretical computer scientists (cf particularly the Logic for Computable Functions [13, Chapter 10]). These systems separate 'partial correctness' (the program gives the correct result if it halts) from proof of halting. The former is completely axiomatised; the latter depends on associating with the state of the computation an element of a well-ordered set which decreases as the computation proceeds, and so it depends on the undecidable judgement that a set is well-ordered. This splitting of the theory into a decidable and an undecidable part is in the same spirit as my decidable 'derivation tree' predicate and undecidable 'validity', below. The usual correctness calculi are not designed with foundational questions in mind so they cannot be used as they stand; however I do borrow some techniques, notably the notion of a 'least fixed point' operator to express recursion. Rules for induction on computations (without mentioning numbers) can be given in an extremely elegant form in terms of this operator (see §1.5).

Thus protologic is based on schematic arguments and induction. Open-endedness is embodied most clearly in the question of which functions are total. Consider the class of sequents

$$(x)\ \mathrm{number}(x) \longrightarrow \mathrm{number}(fx)$$

for functions f. Because we can enumerate derivation trees we can enumerate f's for which the above sequent is derivable, ie f's provably total as number-theoretic functions. These will include the primitive recursive functions (I prove this in §1.6), and probably no others. Now define a recursive 'universal' function F by

$$F(x,y) \equiv \begin{cases} \text{the } x\text{'th provably total function applied to } y & \text{(if } x \text{ is a number)} \\ \text{undefined} \quad \text{(otherwise).} \end{cases}$$

Clearly, (x) number(x) $\longrightarrow$ number(F(x,x)+1) is true (informally) but not derivable, for (λx.F(x,x)+1) is not one of the f's.

More generally, given any decidable class of functions which we have proved total (in some sense) we can diagonalise out of it; and by iterating this 'transcendence' process in various ways (as in §0.4) we generate the open-ended universe of functions which we would informally accept as total. Totality questions are important because in protologic everything is proved relative to the successful halting of some other process.

We can embody this in an 'ordinal logic' approach. Call a derivation in protologic using just schematic arguments and induction a 'derivation of <u>level 0</u>'; a derivation of <u>level 1</u> is one that allows as an extra axiom any reflection principle of the form

$$(\underline{x})\ D[\ \longrightarrow X] \longrightarrow X,$$

where X is a term, D is a derivation tree of level 0 (both possibly containing $\underline{x}$ as free variables), and D[$\longrightarrow$ X] means D is a derivation tree of the sequent $\longrightarrow$ X. A derivation tree of <u>level 2</u> is defined similarly except that D may now be of level 1. And so on, for all finite levels. A derivation tree of <u>level ω</u> allows reflection principles with D of any finite level. We can continue in this way to define levels for all constructive ordinals.

Generally I shall allow in derivation trees reflection principles for *any* D. This means that some derivations will be unsound (if, say, a tree uses a reflection principle with itself as 'D'). A derivation tree D with reflection priciples

$$(\underline{x})\ D_i[\ \longrightarrow X_i] \longrightarrow X_i$$

$(i = 1, \ldots k)$ is said to __depend on__ the set of trees $\{D_i \mid \text{all } \underline{x}, \text{ all } i\}$. Associate with any derivation tree D the tree consisting of D, the trees it depends on, the trees *they* depend on, etc. (this is a tree with trees as nodes); call a tree __valid__ iff its associated tree is well-founded (roughly, iff it has a constructive ordinal as a level). Clearly a derivation is informally sound iff it is valid in this sense. Because well-foundedness is not recursive this is the undecidable component of protologic: the ordinal hierarchy of derivation tree levels represents the open-ended universe of valid mathematical arguments.

Using the definitions (α)-(δ),(ε'),(ζ') of §1.0 and §1.2 we can then define the proof relation $\vdash$ in terms of protologic. The question of validity will arise for proofs as well: roughly, a object will be a __valid proof__ iff the protological derivations referred to in it are valid as derivation trees. For example, in the implication clause (ζ'), a valid proof of $A \supset B$ will contain a mapping from valid proofs of A to valid proofs of B, together with valid protological 'evidence' that the mapping works. This enables us to define 'valid proof of $A \supset B$' in terms of 'valid proof of A', 'valid proof of B' and 'valid protological derivation'. Intuitionistic proof is separated into a decidable component ($\vdash$) and an 'intuitive' component (validity), the latter being due to the undecidability of well-foundedness.

§1.5 The protological sequent calculus

In this section I shall give the formalism for protologic as a sequent calculus. First we need to specify precisely the universe of mathematical objects we are talking about.

An <u>object</u> is either a basic object (True,False,nil,0,1,2,3,... enumerated in that order), a pair of objects, or a partial recursive function from objects to objects. Objects can be coded as numbers: object number n is

(i) (if $n \equiv_3 0$) basic object no. n/3 (counting True as no. 0),

(ii) (if $n \equiv_3 1$) the pair (object no. $\pi_L(\frac{n-1}{3})$, object no. $\pi_R(\frac{n-1}{3})$),

(iii) (if $n \equiv_3 2$) the function: (object no. m) $\mapsto$ (object no. $\left\{\frac{n-2}{3}\right\}(m)$),

where π_L and π_R are primitive recursive pair projection functions on numbers and {p} is the p'th partial recursive function in a fixed coding.

The purpose of describing the coding was to give a more concrete view of the object universe. Anyone who considers my objects too abstract or unclear is welcome to translate all arguments about them into arguments about code numbers. That is to say, in all that follows replace 'object' by 'number', regard all variables as ranging over numbers, translate 'function f applied to a' as '$\left\{\frac{f-2}{3}\right\}(a)$', translate 'the basic object True' as 0, 'the basic object 2' as 15, etc., according to the coding.

Notice that functions are given intensionally, as algorithms rather than sets of ordered pairs; they should be thought of concretely as programs or Turing machines. Two programs may give the same output for all inputs and yet be different programs (ie different objects) if they work out the output in different ways: a program is a string of characters conforming to the syntax of a programming language, and even if two programs differ only in the names they give to variables or the presence of a redundant statement (eg 'Set x equal to x') they are none the less distinct. In terms of code numbers, {17} may be extensionally equivalent to {25}, but however uninterested we may be in the distinction between them we are not at liberty to 'identify' 17 with 25, or object number 3x17+2 with object number 3x25+2.

Define three objects, = , S and fxpt , as follows:

= is equality of objects (ie of their code numbers): it maps any object of the form (a,a) to True and any other object to False.

S is a successor function: it is defined on all objects, is injective, maps nothing to 0, 0 to 1, 1 to 2, etc.

fxpt is a function which takes an object argument f; if f is not a function the result is nil; otherwise the result is a function g such that

$$\text{for all } x: \qquad gx = (fg)x;$$

ie g is a fixed-point of f (extensionally speaking). The Recursion Theorem guarantees the existence of a g for every f: in fact, the usual proof of the theorem yields a function for obtaining g from f, and it is this function that I call "fxpt". The g so obtained will actually be the *least* fixed-point of f; ie any other fixed-point is an extension of it.

The basic object nil is used to indicate an empty list and as the result of a 'failed' computation.

To define protologic I need a language for systematically referring to objects. I shall use letters and other character combinations as variables and constants (denoting objects), and as metavariables (denoting terms or variables). Examples of constants are '0', 'nil' and 'fxpt'. Only variables will actually appear in the terms of the language. I shall never actually write down a term or a variable: instead I shall refer to them by names. When I write an expression that looks like a term except that it contains constants or metavariables it is a name for the term obtained by substituting for each constant or metavariable the object, term or variable it denotes; for example if X = '....' and Y = '----' are terms then (X,Y) = '(....,----)'. When distinct metavariables denoting variables appear in the same expression they denote distinct variables.

Now define a <u>term</u> as

(i) an object (or, strictly, some syntactic representation of an object, since terms ought to be syntactic entities; for definiteness, let us use the code numeral of the object);

(ii) a variable;

(iii) $(\lambda(x_1,\ldots x_k).T)$ where $x_1,\ldots x_k$ are variables and T is a term; for k=1 this is written $(\lambda x_1.T)$;

(iv) (a,b) where a and b are terms;

(v) (ab) where a and b are terms;

(vi) (if a then b else c) where a,b and c are terms.

As informal metanotation, introduce n-tuples: $(a,b,\ldots d,e) \equiv (a,(b,\ldots(d,e)\ldots))$; $(a) \equiv a$. Also introduce substitution notation: $T\begin{bmatrix}T_1 & & T_k\\ x_1 & ,\ldots & x_k\end{bmatrix}$ means T with $T_1,\ldots T_k$ substituted simultaneously for free occurrences of $x_1,\ldots x_k$ respectively (after first renaming any bound variable of T which occurs free in $T_1,\ldots T_k$). I shall use underlined metavariables to indicate lists of distinct variables.

The semantics of terms is given by an interpretation mapping i, a partial function mapping a term without free variables to the object it is intended to denote; it is defined by the Recursion Theorem, as follows.

For T = (the code numeral of) an object, iT is the object.

For T = $(\lambda(x_1,\ldots x_k).U)$, iT is the function f defined as follows. (Let $y_1,y_2,\ldots$ be an infinite list of variables fixed once and for all; let $V = U\begin{bmatrix} y_1 \ldots y_k \\ x_1 \ldots x_k \end{bmatrix}$.) Now, f takes an argument object; if it is not a k-tuple it gives the result nil; if it is it splits it into k components, substitutes them for $y_1,\ldots y_k$ in V, then applies i to the term to give its result. (The purpose of going via $y_1,\ldots y_k$ is to ensure that $i(\lambda(x_1,\ldots x_k).U) = f = i(\lambda(z_1,\ldots z_k).U\begin{bmatrix} z_1, \ldots z_k \\ x_1, \ldots x_k \end{bmatrix})$ for any variables $z_1,\ldots z_k$ (thus since 'x_1',...'x_k' are metavariables denoting variables I shall never need to specify *which* variables).)

For T = (A,B), iT = (iA,iB).

For T = (AB), $iT = \begin{cases} (iA)(iB) & \text{if } iA \text{ is a function} \\ \text{undefined} & \text{otherwise} \end{cases}$.

For T = (if A then B else C) , i first evaluates iA; then

$$iT = \begin{cases} iB & \text{if } iA = \text{True} \\ iC & \text{if } iA = \text{False.} \\ \text{undefined} & \text{otherwise} \end{cases}$$

This completes the semantics of terms. Notice how this definition is unusual. The usual way of defining an interpretation on terms would be to define iT in terms of iA,...iB where A,...B are the subterms of T.

In my clause for λ-terms above, iT depends on the *way* iV is calculated. As remarked above, extensionally equivalent functions are not necessarily the same object, so $i(\lambda x.X) \neq i(\lambda x.Y)$ even if $iX = iY$ for all x, unless X and Y are identical terms: $(\lambda x.T)$ denotes a program to work T out, and this is a different program for each T. So $i(\lambda(\underline{x}).T)$ may be regarded as a 'quoted' form of T (where $\underline{x}$ are the free variables in T or any variable if T has no free variables), ie a representation of T as an object from which T is effectively recoverable. Also, in the clause for 'if' terms, if iA = True then iC is never calculated, so that iT may be defined even if iC (or iB in the case where iA = False) isn't (ie if the computation doesn't halt). Thus i must be defined in terms of its intension (code number), not just its values on the subterms: the Recursion Theorem permits this. It is clear that my terms are not much like the terms of λ-calculus; they are more like expressions in a functional programming language. Indeed, my term language could almost be regarded as a dialect of LISP. Some definitions and metanotation follow.

If I write something like "define X by $X(\underline{x}_1)(\underline{x}_2)\ldots(\underline{x}_k) \equiv T$" I mean that 'X' is short for $(\lambda(\underline{x}_1).(\lambda(\underline{x}_2)\ldots(\lambda(\underline{x}_k).T)\ldots))$ if 'X' doesn't occur in T, or $(\text{fxpt } (\lambda f.(\lambda(\underline{x}_1)\ldots(\lambda(\underline{x}_k).T(^{f}_{X}))\ldots)))$ if 'X' does occur in T.

Write ((ab)c) as (abc) , ((abc)d) as (abcd) , etc.; omit the outside brackets round whole terms of the form (ab...d); write $(\lambda w.(\lambda x.\ldots(\lambda z.T)\ldots))$ as $(\lambda wx\ldots z.T)$; write $\bar{t}$ for the constant function defined by $\bar{t}x \equiv t$ (where x is any variable not occurring in

the term t; it makes no difference to $\bar{t}$ which x is chosen);
for any f and g define $(f \circ g)x \equiv f(gx)$; write $=(a,b)$ as $a=b$;
define pair projection functions $\pi_0(x,y) \equiv x$, $\pi_1(x,y) \equiv y$;
define 'not' by not $x \equiv (x=\text{False})$.

Introduce four new objects, <u>parts</u>, <u>#vbls</u>, <u>const</u> and <u>subst</u>. The first three analyse the intension of functions; each is a function that gives the result nil if its argument is not the interpretation of a term of the form $(\lambda(\underline{x}).T)$. From the definition of i it is clear that objects which are $i(\lambda(\underline{x}).T)$ for some T have a characteristic form: they are functions which substitute their input in a term and apply i to it. Thus it is decidable whether a given object is $i(\lambda(\underline{x}).T)$ for any T, and if so we can effectively extract T (up to renaming of the variables $\underline{x}$).

The function <u>parts</u> maps $i(\lambda(\underline{x}).T)$ to an object which indicates the structure of T and its subterms; eg if $T = (A,B)$ then (parts $(\lambda(\underline{x}).T)) \equiv (3,(\lambda(\underline{x}).A),(\lambda(\underline{x}).B))$, where the 3 indicates that T is a pair and $(\lambda(\underline{x}).A)$, $(\lambda(\underline{x}).B)$ are representations of the subterms A,B as objects; as another example, if T is a variable, say the i'th in the list $\underline{x}$, the value is (1,i), where the 1 indicates that T is a variable. The full definition is given below in the conversion rules.

<u>#vbls</u> maps $i(\lambda(x_1,\dots x_k).T)$ to k.

<u>const</u> maps $i(\lambda(\underline{x}).T)$ to True if $\underline{x}$ don't occur free in T, to False otherwise; ie it determines whether its argument is a constant function.

The final function <u>subst</u> substitutes a term t (given in 'quoted' form as $\bar{t}$) in a λ-term $(\lambda x.T)$ to give $\overline{T(\frac{t}{x})}$; it gives nil for arguments not of the right form.

A <u>sequent</u> is an expression of the form

$$(\underline{x})\ T_1, T_2, \ldots T_k \longrightarrow T_0$$

where $\underline{x}$ is a (possibly empty) list of variables and $T_0, T_1, \ldots T_k$ are terms with free variables from $\underline{x}$; k may be 0, so that $(\underline{x}) \longrightarrow T_0$ is a possible sequent. The informal meaning is that, regardless of the objects substituted for $\underline{x}$, if $T_1, \ldots T_k$ denote True so does T_0.

Next I want to define 'conversion' between terms; ie a relation such that 'T conv T′' means it follows from the forms of T and T′ that iT = iT′ (if either is defined, when arbitrary objects are substituted for the free variables). I cannot allow the general λ-conversion rule '$(\lambda x.T)A$ conv $T(\overset{A}{x})$' for two reasons:

(i) Suppose A is undefined (ie i doesn't halt on A). Then $(\lambda x.T)A$ will necessarily be undefined because to work out $i((\lambda x.T)A)$ one has to work out iA. But $T(\overset{A}{x})$ may be defined if x doesn't occur in T, or if T = (if B then C else x) (since, if iB = True, iA will never be worked out in the calculation of $i(T(\overset{A}{x}))$). So for λ-conversion to work I insist that A be 'manifestly well-defined' (ie it is clear from its form that it is defined) or that x 'occur essentially' in T (meaning that, in a calculation of $i(T(\overset{A}{x}))$, iA must be calculated): both these concepts are defined below.

(ii) The second obstacle to λ-conversion is that x may occur 'intensionally' in T, ie within a λ-term. Take $T \equiv (\lambda y.(x,y))$ for example. $i(T(\overset{A}{x}))$ is $i(\lambda y.(A,y))$, a function that, when applied to an argument, will work out iA in the course of its computation. Whereas $i((\lambda x.T)A)$ is $i(\lambda y.(a,y))$, where by 'a' I mean the object iA, assuming A is defined: $i(\lambda y.(a,y))$ doesn't *calculate* 'iA' when applied to an argument but simply uses the *result*. Thus when A is undefined so will be $(\lambda x.T)A$, but $T(\begin{smallmatrix}A\\x\end{smallmatrix})$ is always defined. Thus for λ-conversion I stipulate that x doesn't occur within the scope of a 'λ' in T, or that A is a variable or object (for which case there is no problem).

A term t is <u>manifestly well-defined</u> (M.W.D.) iff t is an object, variable or term of the form $(\lambda(\underline{x}).U)$; or t is (a,b) with a and b M.W.D.; or t is (if a then b else c) with a,b and c M.W.D.; or t is ab where b is M.W.D. and a is =, S, fxpt, parts, #vbls, const, subst or $(\lambda(\underline{x}).U)$ where U is M.W.D..

A variable x <u>occurs essentially</u> in a term T iff T is x; or T is (A,B) or AB where x occurs essentially in A or B; or T is (if X then Y else Z) where x occurs essentially in X or *both* Y and Z.

Define <u>conversion</u> of terms as follows.

X conv X; X conv Y implies Y conv X;

T conv T′, if T and T′ have no free variables and iT = iT′; (Rule (*))

$(\lambda(x_1,\ldots x_k).T)$ conv $(\lambda(z_1,\ldots z_k).T\begin{bmatrix}z_1\\x_1\end{bmatrix},\ldots\begin{bmatrix}z_k\\x_k\end{bmatrix})$;

$(\lambda x_1x_2\ldots x_k.T)t_1t_2\ldots t_k$ conv $(\lambda(x_1,\ldots x_k).T)(t_1,\ldots t_k)$;

$(\lambda(x_1,..x_k).T)(X_1,..X_k)$ conv $T\begin{bmatrix}X_1\\x_1\end{bmatrix},\ldots\begin{bmatrix}X_k\\x_k\end{bmatrix}$ provided, for i=1...k,

x_i occurs essentially in T or X_i is M.W.D.; and x_i must not occur free within a λ-term in T;

$(\lambda x.T)y$ conv $T(\begin{smallmatrix}y\\x\end{smallmatrix})$, where y is a variable or object;

$(\lambda(\underline{x}).T)t$ conv $(\lambda(\underline{x}).T')t$, if T conv T′;

$$\left.\begin{array}{l}(A,B) \text{ conv } (A',B')\\ (AB) \text{ conv } (A'B')\\ (\text{if } A \text{ then } B \text{ else } C) \text{ conv } (\text{if } A' \text{ then } B' \text{ else } C')\end{array}\right\}\text{ where }\left\{\begin{array}{l}A \text{ conv } A'\\ B \text{ conv } B';\\ C \text{ conv } C'\end{array}\right.$$

fxpt ϕ X conv ϕ(fxpt ϕ) X;

$SX=0$ conv $\overline{\text{False}}\ X$;

$SX=SY$ conv $X=Y$;

$X=Y$ conv $Y=X$;

$X=X$ conv $\overline{\text{True}}\ X$;

True=False conv False;

(if True then X else Y) conv X;

(if False then X else Y) conv Y;

R(if Z then X else Y) conv (if Z then RX else RY);

(if X=Y then A else A) conv $\bar{A}(X,Y)$;

parts($\lambda(\underline{x})$.a) conv (0,a), if a is an object, or variable not in $\underline{x}$;

parts($\lambda(x_1,\ldots x_k).x_i$) conv (1,i) $(1 \leq i \leq k)$;

parts($\lambda(\underline{x}).(\lambda(\underline{y}).U)$) conv (2,($\lambda(\underline{x},\underline{y})$.U));

parts($\lambda(\underline{x})$.(A,B)) conv (3,($\lambda(\underline{x})$.A),($\lambda(\underline{x})$.B));

parts($\lambda(\underline{x})$.AB) conv (4,($\lambda(\underline{x})$.A),($\lambda(\underline{x})$.B));

parts($\lambda(\underline{x})$.if A then B else C) conv (5,($\lambda(\underline{x})$.A),($\lambda(\underline{x})$.B),($\lambda(\underline{x})$.C));

#vbls($\lambda(x_1,\ldots x_k)$.T) conv k;

const($\lambda(\underline{x})$.T) conv $\begin{cases} \text{True}, & \text{if } \underline{x} \text{ don't occur free in T} \\ \text{False}, & \text{if } \underline{\text{some}} \text{ of } \underline{x} \text{ occur free in T} \end{cases}$;

subst $\bar{t}$ (λx.T) conv $\overline{T(^{t}_{x})}$.

This completes the definition of conversion. The sequent calculus follows. Let Γ and Δ be arbitrary, possibly empty, lists of terms.

Axioms

($\underline{x}$) $\rightarrow$ True

($\underline{x}$) False $\rightarrow$ T

($\underline{x}$) T $\rightarrow$ T=True

($\underline{x}$) A $\rightarrow$ B (where A conv B)

($\underline{x}$) U=V, $A(^{U}_{x}) \rightarrow A(^{V}_{x})$ (x not occurring free within a λ-term of A)

Rules

Exchange:
$$\frac{(\underline{x})\ \Gamma,\ A,\ B,\ \Delta \rightarrow C}{(\underline{x})\ \Gamma,\ B,\ A,\ \Delta \rightarrow C} \qquad \frac{(\underline{x})\ \Gamma \rightarrow C}{(\underline{z})\ \Gamma \rightarrow C}$$

(where $\underline{z}$ is a permutation of the variables $\underline{x}$)

Contraction:
$$\frac{(\underline{x})\ \Gamma,\ A,\ A \rightarrow C}{(\underline{x})\ \Gamma,\ A \rightarrow C}$$

Weakening:
$$\frac{(\underline{x})\ \Gamma \rightarrow C}{(\underline{x})\ \Gamma,\ A \rightarrow C}$$

Cut:
$$\frac{(\underline{x})\ \Gamma,\ A \rightarrow B \qquad (\underline{x})\ \Delta \rightarrow A}{(\underline{x})\ \Gamma,\ \Delta \rightarrow B}$$

Redundant variable:
$$\frac{(\underline{x},y)\ \Gamma \rightarrow C}{(\underline{x})\ \Gamma \rightarrow C}$$
and vice versa

(where y doesn't occur free in Γ or C)

Instantiation:
$$\frac{(\underline{x},y)\ A_1y,\ldots A_ky \rightarrow By}{(\underline{x})\ A_1t,\ldots A_kt \rightarrow Bt} \qquad (k \geq 1).$$

If:
$$\frac{(\underline{x})\ \Gamma,\ T,\ A \rightarrow C \qquad (\underline{x})\ \Gamma,\ \text{not } T,\ B \rightarrow C}{(\underline{x})\ \Gamma,\ \text{if } T \text{ then } A \text{ else } B \rightarrow C}$$

Fxpt: (a)
$$\frac{(\underline{x},\underline{z})\ \Phi Y\ (\underline{x}) \rightarrow Y(\underline{x})}{(\underline{x},\underline{z})\ \text{fxpt}\ \Phi\ (\underline{x}) \rightarrow Y(\underline{x})}$$

(b)
$$\frac{(F_1,\ldots F_k,\ \underline{x},\ \underline{z})\ X(\Phi_1F_1,\ldots\Phi_kF_k)(\underline{x}) \rightarrow \Psi(X(F_1,\ldots F_k))(\underline{x})}{(\underline{x},\ \underline{z})\ X(\text{ fxpt } \Phi_1,\ \ldots \text{fxpt } \Phi_k\ )(\underline{x}) \rightarrow \text{fxpt } \Psi\ (\underline{x})}$$

(where Φ, $\Phi_1,\ldots\Phi_k$, Ψ are of the form $(\lambda F.\ (\lambda(\underline{x}).\ \text{if } C(\underline{x}) \text{ then } R(\underline{x})$ else $F(H(\underline{x}))))$, where C,R,H and Y are terms that may contain $\underline{z}$.

X is a term of the form

$$(\lambda(y_1,\dots y_k).\ (\lambda(\underline{x}).\ F(y_1G_1,\dots y_kG_k)))$$

where F, $G_1,\dots G_k$ may contain $\underline{x}$, $\underline{z}$.)

This completes the list of axioms and rules of the sequent calculus. The fxpt rules embody induction on the length of the computation in the antecedent of the conclusion. A function fxpt Φ , for Φ of the form specified in the fxpt rules, works by applying H to its argument $\underline{x}$ repeatedly until it satisfies C, then applying R to give the final result. For example, in Fxpt Rule (a) the premise splits (by the If Rule) into two cases, $(\underline{x},\underline{z})$ $C(\underline{x})$, $R(\underline{x}) \longrightarrow Y(\underline{x})$ and $(\underline{x},\underline{z})$ not $C(\underline{x})$, $Y(H(\underline{x})) \longrightarrow Y(\underline{x})$, which imply that $Y(\underline{x})$ holds for the iterate $\underline{x} = H^n(\underline{y})$ of the original argument $\underline{y}$ at which $C(\underline{x})$ first holds and that the property Y is inherited up the sequence of iterates from $\underline{x}$ to $\underline{y}$. For more about the properties of fxpt and the fxpt rules, see [13], Chapter 4, particularly Corollary 4.3 (Park's Theorem).

The axioms and rules are sound in terms of the informal protologic discussed in §1.4: ie if $(\underline{x})$ A,...B $\longrightarrow$ C is derivable and, for particular objects substituted for $\underline{x}$, A,...B denote True, then so does C. Completeness is impossible to establish because it involves comparing the precise sequent calculus (with reflection principles added) with the informal notion of protologic. From extensive experience with the sequent calculus I am convinced that it is strong enough for protologic. However, I do not insist on this. In future it may prove necessary, say, to strengthen the fxpt rules to allow more elaborate kinds of induction, or the conversion rules; I do not think this will happen, but if it does then all the arguments of the following sections will still work.

§1.6 Some derivations

First define a truth function & by

$$\&AB \equiv (\text{if } A{=}\text{True then } B \text{ else } A),$$

and write &AB as A&B. It is trivial to verify

$\dfrac{(\underset{\sim}{x})\ \Gamma,\ A, \ldots C \longrightarrow D}{(\underset{\sim}{x})\ \Gamma,\ A\&\ldots\&C \longrightarrow D}$ and vice versa; $\dfrac{(\underset{\sim}{x})\ \Gamma \longrightarrow A\ \ldots\ (\underset{\sim}{x})\ \Delta \longrightarrow C}{(\underset{\sim}{x})\ \Gamma, \ldots \Delta \longrightarrow A\&\ldots\&C}$;

where the brackets in A&...&C may occur anywhere.

Next define a term $\perp$ describing a non-halting computation, eg

$$\perp \equiv \text{fxpt } \Phi\ 0, \qquad \text{where } \Phi Xx \equiv Xx.$$

Then $(\underset{\sim}{x})\ F\perp \longrightarrow A$ is derivable, for any terms F and A.

Now I shall derive the protological induction rule used in §2.6 for Rule (δ) (Induction) of Heyting Arithmetic. Define 'num x' to mean x is a number; ie

$$\text{num } x \equiv \text{fxpt } \Phi_N(x,0),$$

where $\Phi_N X(x,i) \equiv (\text{if } x{=}i \text{ then True else } X(x,Si))$.

Then the induction rule to be derived is

$$\frac{\longrightarrow A0 \qquad\qquad (x)\ Ax \longrightarrow A(Sx)}{(x)\ \text{num } x \longrightarrow Ax}.$$

Now, the conclusion can be derived by Cut from

$$\begin{cases} (x)\ \text{num } x \longrightarrow B'x & \text{(i)} \\ (x)\ B'x \longrightarrow Bx & \text{(ii)} \\ (x)\ Bx \longrightarrow Ax & \text{(iii)} \end{cases}$$

where $Bx \equiv \text{fxpt } \Psi(x,0)$, $B'x \equiv \text{fxpt } \Psi'(x,0,0)$,

$$\Psi X(x,i) \equiv \text{if } x=i \text{ then } Ai \text{ else } X(x,Si),$$

$$\Psi' X(x,i,j) \equiv \text{if } x=i \text{ then } Aj \text{ then } X(x,Si,Sj).$$

(i) may be rewritten

$(x)\ \text{fxpt}\ \Phi_N(x,0) \longrightarrow \text{fxpt}\ \Psi'(x,0,0),$

which is derived by Instantiation from

$(x,i)\ \text{fxpt}\ \Phi_N(x,i) \longrightarrow \text{fxpt}\ \Psi'(x,i,0),$

which is derived by Fxpt Rule (a) from

$(x,i)\ \Phi_N(\lambda(x,i).\ \text{fxpt}\ \Psi'(x,i,0))(x,i) \longrightarrow \text{fxpt}\ \Psi'(x,i,0),$

which splits, by the If Rule, into two:

$$\begin{cases}(x,i)\ x=i,\ \text{True} \longrightarrow A0,\\ (x,i)\ x\neq i,\ \text{fxpt}\ \Psi'(x,Si,0) \longrightarrow \text{fxpt}\ \Psi'(x,i,0),\end{cases}$$

where $x\neq i$ is short for not($x=i$). The first sequent follows from the premise $\longrightarrow A0$; for the other one the RHS converts to fxpt $\Psi'(x,Si,S0)$ (under the hypothesis $x\neq i$), and then the sequent is derived by Instantiation and Weakening from

$(x,k,j)\ \text{fxpt}\ \Psi'(x,k,j) \longrightarrow \text{fxpt}\ \Psi'(x,k,Sj),$

which is derived by Fxpt Rule (a) from

$(x,k,j)\ \Psi'(\lambda(x,k,j).\ \text{fxpt}\ \Psi'(x,k,Sj))(x,k,j) \longrightarrow \text{fxpt}\ \Psi'(x,k,Sj).$

This splits into two cases by the If Rule,

$$\begin{cases}(x,k,j)\ x=k,\ Aj \longrightarrow A(Sj),\\ (x,k,j)\ x\neq k,\ \text{fxpt}\ \Psi'(x,Sk,S(Sj)) \longrightarrow \text{fxpt}\ \Psi'(x,Sk,S(Sj));\end{cases}$$

the second is trivial and the first follows from the other premise

$$(x)\ Ax \longrightarrow A(Sx).$$

This completes (i); (ii) may be rewritten

$(x)\ \text{fxpt}\ \Psi'(x,0,0) \longrightarrow \text{fxpt}\ \Psi(x,0),$

which is derived by Instantiation from

$(x,i)\ \text{fxpt}\ \Psi'(x,i,i) \longrightarrow \text{fxpt}\ \Psi(x,i),$

which is derived by Fxpt Rule (b) from

$(x,i,F)\ \Psi' F(x,i,i) \longrightarrow \Psi(\lambda(x,i).\ F(x,i,i))(x,i),$

which splits by the If Rule into

$$\begin{cases}(x,i,F)\ x{=}i,\ Ai \longrightarrow Ai, \\ (x,i,F)\ x{\neq}i,\ F(x,Si,Si) \longrightarrow F(x,Si,Si),\end{cases}$$

both of which are trivial.

This completes (ii); (iii) may be rewritten

(x) fxpt $\Psi(x,0) \longrightarrow Ax$,

which is derived by Instantiation from

(x,i) fxpt $\Psi(x,i) \longrightarrow Ax$,

which is derived by Fxpt Rule (a) from

$(x,i)\ \Psi(\lambda(x,i).\ Ax)(x,i) \longrightarrow Ax$,

which splits into two cases:

$$\begin{cases}(x,i)\ x{=}i,\ Ai \longrightarrow Ax, \\ (x,i)\ x{\neq}i,\ Ax \longrightarrow Ax,\end{cases}$$

both of which are trivial.

This completes the derivation of the Induction Rule.

The final derivation needed is $(\underline{x})$ num $\underline{x} \longrightarrow$ num$(f\underline{x})$, for primitive recursive f. (This should be regarded as a formalisation of the argument in §0.3.) Here $\underline{x}$ is a list $(x_0,\ldots x_k)$, and num $\underline{x}$ is short for num x_0 &...& num x_k. By 'primitive recursive' I mean f is the zero function, the identity, S, obtained by substitution from primitive recursive functions, or defined from primitive recursive a and g by

$$f(x,\underline{z}) \equiv \text{fxpt}\ \Phi(x,\underline{z},0,a\underline{z}),$$

where $\Phi X(x,\underline{z},i,v) \equiv$ (if $x{=}i$ then v else $X(x,\underline{z},Si,g(i,\underline{z},v))$).

(f so defined will satisfy $\begin{cases}f(0,\underline{z}) &= a\underline{z} \\ f(Sx,\underline{z}) &= g(x,\underline{z},f(x,\underline{z}))\end{cases}$.

We can show that f does satisfy these equations, ie we can derive

$$(\underline{z})\ P(f(0,\underline{z})) \longrightarrow P(a\underline{z}), \qquad (\underline{z})\ P(a\underline{z}) \longrightarrow P(f(0,\underline{z})),$$

$$(x,\underline{z})\ P(f(Sx,\underline{z})) \longrightarrow P(g(x,\underline{z},f(x,\underline{z}))),$$

$$(x,\underline{z})\ P(g(x,\underline{z},f(x,\underline{z}))) \longrightarrow P(f(Sx,\underline{z})),$$

for arbitrary P; in the first two cases use '$f(0,\underline{z})$ conv $a\underline{z}$'; the other two follow by derivations similar to, but simpler than, the other derivations in this section.)

The derivation is by induction on the construction of f. For the zero and identity functions it is trivial. For f=S, we want

(x) fxpt $\Phi_N(x,0) \rightarrow$ fxpt $\Phi_N(Sx,0)$,

which is obtained by Instantiation from

(x,i) fxpt $\Phi_N(x,i) \rightarrow$ fxpt $\Phi_N(Sx,i)$,

which derives, by Fxpt Rule (a), from

$(x,i)\ \Phi_N(\lambda(x,i).\text{fxpt}\ \Phi_N(Sx,i))(x,i) \rightarrow$ fxpt $\Phi_N(Sx,i)$,

which splits into two cases:

$$\begin{cases}(x,i)\ Sx=i,\ \Phi_N(\lambda(x,i).\text{fxpt}\ \Phi_N(Sx,i))(x,i) \rightarrow \text{True}\\ (x,i)\ Sx\neq i,\ \Phi_N(\lambda(x,i).\text{fxpt}\ \Phi_N(Sx,i))(x,i) \rightarrow \text{fxpt}\ \Phi_N(Sx,Si)\end{cases}$$

of which the first is trivial and the second further subdivides into

$$\begin{cases}(x.i)\ Sx\neq i,\ x=i,\ \text{True} \rightarrow \text{fxpt}\ \Phi_N(Sx,Si),\\ (x,i)\ Sx\neq i,\ x\neq i,\ \text{fxpt}\ \Phi_N(Sx,Si) \rightarrow \text{fxpt}\ \Phi_N(Sx,Si).\end{cases}$$

In the first case, the condition x=i can be converted to Sx=Si, so that the RHS becomes True, so the sequent is derivable. The second case is trivial. This completes the derivation for the case f=S.

Now suppose f is obtained by substitution from primitive recursive functions:

$$f\underline{x} \equiv g(h_1\underline{x},\ldots h_k\underline{x}),$$

where $h_1,\ldots h_k$ need not be functions of all the variables $\underline{x}$. Then

$(\underline{x})$ num $\underline{x} \rightarrow$ num $(f\underline{x})$ is derivable by Cut from

$$\begin{cases}(x)\ \text{num}\ \underline{x} \rightarrow \text{num}\ (h_1\underline{x}),\\ \ldots\ldots\ldots\ldots\ldots\ldots\\ (\underline{x})\ \text{num}\ \underline{x} \rightarrow \text{num}\ (h_k\underline{x}),\\ (\underline{x})\ \text{num}\ (h_1\underline{x}),\ldots\text{num}\ (h_k\underline{x}) \rightarrow \text{num}\ (g(h_1\underline{x},..h_k\underline{x})).\end{cases}$$

The last sequent derives by Instantiation from

$(y_1,\ldots y_k)$ num $y_1,\ldots$num $y_k \rightarrow$ num $(g(y_1,\ldots y_k))$.

So $(\underline{x})$ num $\underline{x} \rightarrow$ num $(f\underline{x})$ is derivable from the corresponding sequents for g and $h_1,\ldots h_k$, as required.

Finally, suppose f is defined by primitive recursion from a and g. We want to derive

$$\frac{(\underline{z})\ \text{num}\ \underline{z} \to \text{num}\ (a\underline{z}) \qquad (i,\underline{z},v)\ \text{num}\ i,\ \text{num}\ \underline{z},\ \text{num}\ v \to \text{num}\ (g(i,\underline{z},v))}{(x,\underline{z})\ \text{num}\ x,\ \text{num}\ \underline{z} \to \text{num}\ (f(x,\underline{z}))}.$$

First some auxiliary definitions:

$$\Phi' X(x,\underline{z},i,v) \equiv (\text{if } x{=}i \text{ then } \text{num}\ v \text{ else } X(x,\underline{z},Si,g(i,\underline{z},v))),$$

$$\pi(x,\underline{z},i,v) \equiv (\text{if } \text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v \text{ then } (x,i) \text{ else } \bot).$$

Now, $(x,\underline{z})\ \text{num}\ x,\ \text{num}\ \underline{z} \to \text{num}\ (f(x,\underline{z}))$ is derivable by Cut from:

$$\begin{cases}
(x,\underline{z}) \quad \to \text{num}\ 0 \\
(x,\underline{z})\ \text{num}\ \underline{z} \to \text{num}\ (a\underline{z}) \\
(x,\underline{z})\ \text{num}\ x \to \text{fxpt}\ \Phi_N\ (x,0) \\
(x,\underline{z})\ \text{num}\ \underline{z},\ \text{num}\ 0,\ \text{num}\ (a\underline{z}),\ \text{fxpt}\ \Phi_N\ (x,0) \to \text{fxpt}\ \Phi_N(\pi(x,\underline{z},0,a\underline{z})) \\
(x,\underline{z})\ \text{fxpt}\ \Phi_N\ (\pi(x,\underline{z},0,a\underline{z})) \to \text{fxpt}\ \Phi'\ (x,\underline{z},0,a\underline{z}) \qquad \text{(iv)} \\
(x,\underline{z})\ \text{fxpt}\ \Phi'\ (x,\underline{z},0,a\underline{z}) \to \text{num}\ (\text{fxpt}\ \Phi\ (x,\underline{z},0,a\underline{z})) \qquad \text{(v)} \\
(x,\underline{z})\ \text{num}\ (\text{fxpt}\ \Phi\ (x,\underline{z},0,a\underline{z})) \to \text{num}\ (f(x,\underline{z}))
\end{cases}$$

Everything above is either essentially a premise or trivial, except for (iv) and (v). (iv) may be obtained by Instantiation from

$(x,\underline{z},i,v)\ ((\text{fxpt}\ \Phi_N)\circ\pi)(x,\underline{z},i,v) \to \text{fxpt}\ \Phi'\ (x,\underline{z},i,v);$

which follows by Fxpt Rule (b) from

$(F,x,\underline{z},i,v)\ ((\Phi_N F)\circ\pi)(x,\underline{z},i,v) \to \Phi'(F\circ\pi)(x,\underline{z},i,v).$

Now $((\Phi_N F)\circ\pi)(x,\underline{z},i,v)$ converts to (if $\text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v$ then $\Phi_N F(x,i)$ else $\Phi_N F\bot$. So the sequent splits into two cases

$$\begin{cases}
(F,x,\underline{z},i,v)\ \text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v,\ \Phi_N F(x,i) \to \Phi'(F\circ\pi)(x,\underline{z},i,v) \\
(F,x,\underline{z},i,v)\ \text{not}\ (\text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v),\ \Phi_N F\bot \to \Phi'(F\circ\pi)(x,\underline{z},i,v)
\end{cases}$$

of which the second is trivial because of the $\bot$. The first further subdivides into two cases

$$\begin{cases}
(F,x,\underline{z},i,v)\ x{=}i,\ \text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v,\ \text{True} \to \text{num}\ v \\
(F,x,\underline{z},i,v)\ x{\neq}i,\ \text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v,\ F(x,Si) \to (F\circ\pi)(x,\underline{z},Si,g(i,\underline{z},v))
\end{cases}$$

The first sequent is trivial. The second is derived by Cut from

$$\begin{cases}
(F,x,\underline{z},i,v)\ \text{num}\ \underline{z}\ \&\ \text{num}\ i\ \&\ \text{num}\ v \to \text{num}\ (g(i,\underline{z},v)), \\
(F,x,\underline{z},i,v)\ \text{num}\ i \to \text{num}\ (Si), \\
(F,x,\underline{z},i,v)\ \text{num}\ \underline{z},\ \text{num}\ (Si),\ \text{num}\ (g(i,\underline{z},v)),\ F(x,Si) \to \\
\qquad F(\pi(x,\underline{z},Si,g(i,\underline{z},v))).
\end{cases}$$

The first sequent is essentially a premise. The second is derived above (apart from the redundant variables). The third is trivial, using the definition of $\bar{v}$.

It only remains to derive (v), which is obtained by Instantiation from

$(x,\underline{z},i,v)$ fxpt Φ' $(x,\underline{z},i,v) \rightarrow (\text{numo}(\text{fxpt } \Phi))(x,\underline{z},i,v)$.

This follows by Fxpt Rule (a) from

$(x,\underline{z},i,v)$ $\Phi'(\text{numo}(\text{fxpt } \Phi))(x,\underline{z},i,v) \rightarrow (\text{numo}(\text{fxpt } \Phi))(x,\underline{z},i,v)$,

which splits into two cases

$$\begin{cases}(x,\underline{z},i,v)\ x{=}i,\ \text{num } v \rightarrow (\text{numo}(\text{fxpt } \Phi))(x,\underline{z},i,v)\\ (x,\underline{z},i,v)\ x{\neq}i,\ (\text{numo}(\text{fxpt } \Phi))(x,\underline{z},Si,g(i,\underline{z},v)) \rightarrow (\text{numo}(\text{fxpt } \Phi))(x,\underline{z},i,v)\end{cases}$$

both of which are easy: in the first sequent the RHS converts to num v, and in the second sequent the RHS converts to

$$\text{num}(\text{fxpt } \Phi\ (x,\underline{z},Si,g(i,\underline{z},v))).$$

This completes the derivation.

§1.7 The 'Derivation Tree' predicate, DT

Because I want the sequent calculus to be able to refer to itself I shall now define a predicate DT(D,A) as a term in the sequent calculus language, meaning that D is a derivation tree for the sequent A.

First we need to code sequents and derivation trees as objects. Code the sequent $(x_1,\ldots x_n)\ T_1,\ldots T_k \rightarrow T_0$ as the object $(n,A_1,\ldots A_k,A_0)$ where, for $i = 0,\ldots k$, A_i is the object denoted by $(\lambda x_1x_2\ldots x_n.T_i)$ if n > 0, $\overline{T_i}$ otherwise. For convenience identify a sequent with its code object. A derivation tree will be represented as

$(A,(T_0,T_1))$ or $(A,(T,\text{nil}))$ or (A,nil)

where A is the root node (the conclusion sequent), and the subtrees descending from A are T_0 and T_1, or just T, or none at all, respectively. Define a function <u>#desc</u>, using π_0 and π_1, mapping a tree to the number of subtrees descending from its root.

A <u>sequent schema</u> is a term which is like the representation of a sequent except that it may contain free variables. A <u>derivation tree schema</u> is a derivation made up of sequent schemata instead of sequents, but still conforming to the sequent calculus of §1.5.

To embody the open-endedness considerations of §1.4 I will define DT to allow as an axiom any <u>reflection principle</u>:

$$(\underline{x})\ DT(D, (\ \rightarrow T)) \rightarrow T$$

for any terms D and T.

DT is defined by means of defining equations, as explained in §1.5. Define predicates $inf_1(B,A)$, $inf_2(B,C,A)$, axiom A, meaning $\frac{B}{A}$, $\frac{B\quad C}{A}$ are protological inference steps, and A is an axiom, respectively (allowing reflection principles as axioms). Inf_1, inf_2 and axiom are defined in terms of #vbls, parts, const, and subst; this implies that, for sequent (schemata) A,B,C involving free variables, expressed explicitly as terms in the above coding, $inf_1(B,A)$, $inf_2(B,C,A)$ and axiom(A) all convert to True if $\frac{B}{A}$, $\frac{B\ C}{A}$ are correct inference steps, or A is an axiom, regardless of the free variables; this works since the essential properties of #vbls, etc., are given as conversion rules, and we can follow the computation of $inf_1(B,A)$, etc., converting the terms to True. Now define:

$$
\begin{aligned}
\text{deriv } D \equiv\ & \text{if } \#\text{desc } D = 0 \text{ then } \text{axiom } (\pi_0 D) \text{ else} \\
& \quad \text{if } \#\text{desc } D = 1 \text{ then } \text{inf}_1(\pi_0(\pi_0(\pi_1 D)),\ \pi_0 D)\ \&\ \text{deriv } (\pi_0(\pi_1 D)) \\
& \quad \qquad \text{else } \text{inf}_2(\pi_0(\pi_0(\pi_1 D)),\ \pi_0(\pi_1(\pi_1 D)),\ \pi_0 D)\ \& \\
& \quad \qquad \qquad \text{deriv } (\pi_0(\pi_1 D))\ \&\ \text{deriv } (\pi_1(\pi_1 D)); \\
DT(D,A) \equiv\ & (\pi_0 D = A)\ \&\ (\text{deriv } D).
\end{aligned}
$$

This defines DT: abbreviate DT(D,A) to D[A] (whenever I use square brackets it will always be with this meaning, apart from references).

Chapter 2: Finitary Mathematics

§2.0 Definition of the proof predicate, $\vdash$

The purpose of this chapter is to use the protologic of the previous chapter to define intuitionistic logic (ie $\vdash$) for finitary mathematics explicitly, and use it to justify Heyting Arithmetic.

Consider formulae of first-order predicate logic. Formulae must be interpreted somehow as objects if we are going to define $\vdash$ formally as a predicate of objects, and hence an object itself. To accomplish this, let '$\wedge$','$\vee$','$\supset$','$\exists$','$\forall$' denote distinct objects (say 0,1,2,3,4), and consider the following alternative notation for formulae:

$\overline{T}$ $\equiv$ the atomic formula T (meaning T denotes True),

$(\wedge,A,B) \equiv A \wedge B$,

$(\vee,\overline{A},\overline{B}) \equiv A \vee B$,

$(\supset,\overline{A},\overline{B}) \equiv A \supset B$,

$(\exists,(\lambda x.A)) \equiv \exists x\ A$,

$(\forall,(\lambda x.A)) \equiv \forall x\ A$.

(I omit negation as it can be defined from $\supset$.) Thus, for example, $(\exists,(\lambda x.(\supset,\overline{(\forall,(\lambda y.\overline{fxy}))},\overline{gx})))$ is simply another way of writing $\exists x\ ((\forall y\ fxy) \supset gx)$. The point of this is that when formulae are so rewritten they become terms and so denote objects, as required. (That is to say, I regard formulae as disguised terms.) It is convenient to allow as a formula any term of this form, and also $(\lambda x.F)t$ where F is a formula. I shall use this alternative notation for formulae, or the usual notation, or a mixture of both, whichever is most convenient in the circumstances.

The reason for the overlining of terms in the $\supset$ and $\vee$ clauses is that I want, eg, $T \supset T$ to be provable even if T is undefined: if I said

'$T \supset T \equiv (\supset,T,T)$' then it wouldn't even be defined in this case, so it couldn't be provable. Similarly, $A \vee B$ should be provable even if B is undefined (by proving A), so I must ensure $A \vee B$ is always defined regardless of whether A and B are.

Now define $\vdash$ by

$$\begin{aligned}
p \vdash \bar{T} &\equiv T; \\
(p,q) \vdash (\wedge,x,y) &\equiv p \vdash x \ \& \ q \vdash y; \\
(i,p) \vdash (\vee,\bar{A},\bar{B}) &\equiv \text{if } i \text{ then } p \vdash A \text{ else } p \vdash B; \\
(n,p) \vdash (\exists,f) &\equiv p \vdash fn; \\
(d,h) \vdash (\forall,f) &\equiv d[(x) \longrightarrow hx \vdash fx]; \\
((g,h),d) \vdash (\supset,\bar{A},\bar{B}) &\equiv d[(Q) \longrightarrow gQ[Q \vdash A \longrightarrow hQ \vdash B]].
\end{aligned}$$

Explanation of notation: p,q,i,n,d,f,g,h,x and y are variables, A,B and T are terms. Define $\pi_2 \equiv \pi_0 \circ \pi_0$, $\pi_3 \equiv \pi_1 \circ \pi_0$: then $\pi_i((x_2,x_3),x_1) \equiv x_i$ for $i=1,2,3$: I am using $((x_2,x_3),x_1)$ as the ordered triple of x_1,x_2,x_3 in the context of the '$\supset$' clause above, with projection functions π_1,π_2,π_3. Abbreviate $\pi_i T$ to T_i.

The above clauses define a term in the sequent calculus, by the recursion theorem as explained in §1.5, and hence an object, $\vdash$.

The defining clauses formalise the informal definitions (α)-(δ), (ϵ'), (ζ') of §§1.0,1.2. The '$\supset$' clause perhaps requires explanation. Naively one would write:

$$(d,h) \vdash (\supset,\bar{A},\bar{B}) \equiv d[(Q)\ Q \vdash A \longrightarrow hQ \vdash B] \qquad (*).$$

However, this is not strong enough: with $(*)$ it would be impossible even to justify $X \supset ((X \supset Y) \supset Y)$. For it does not allow us to assume the validity of Q in proving that $Q \vdash A$ implies $hQ \vdash B$; whereas in the clause given above the derivation tree gQ can assume the validity of Q in proving that $Q \vdash A$ implies $hQ \vdash B$. The given '$\supset$' clause allows us to prove everything provable by (*) and more: further apparent strengthenings by a more complicated clause do not yield a stronger notion of '$\supset$'. The '$\supset$' clause is intended to encompass *all* ways in which one could use protologic to establish that if Q is valid and $Q \vdash A$ then (a function of Q) $\vdash B$ (cf Lemma 7 below). Likewise the '$\forall$' clause is intended to encompass all ways in which one could use protologic to establish that (a function of x) $\vdash fx$ for any x (cf Lemma 5 below).

§2.1 Validity of proofs

As remarked in §1.4, the fact that D[A] doesn't always mean that A is protologically proved, since D is allowed to contain reflection principles which may not themselves be sound: D is a valid tree iff the associated tree (defined in §1.4) is well-founded. Since proofs involve derivation trees a validity question will also arise for proofs: a valid proof is one where all the derivation trees ultimately referred to in it are valid trees. Then an intuitionistic proof of a formula A will be a valid P such that $P \vdash A$.

Validity of proofs means, for example, that if P is intended as a proof of $\forall x\, A$ then P is valid iff P_0 is a valid derivation tree and P_1x is valid for all x. This cannot serve as part of a definition of validity as it stands because it is circular. But in fact the circularity is only apparent: being a valid proof for $\forall x\, A$ is defined in terms of being valid for A and being a valid protological derivation. So we can define a notion of validity by recursion on the structure of the formula being proved, provided the proof does specify the formula.

Probably the conceptually clearest way to think of a proof of $\forall x\, A$ is as a function g (ie a program), with the formula A and the derivation of $(x) \longrightarrow gx \vdash A$ embedded in the program in comment statements (which do not affect the execution of the program but make its operation more transparent). Thinking of protologic as a calculus for program correctness, this is like a program annotated with a correctness proof according to Hoare logic, ie a 'self-documenting' program that says what it does and proves that it does it. This is also reminiscent of Martin-Löf's view (§1.3) that a function is *manifestly* of a certain 'type', so that the proof of $\forall x\, A$ is just the function; perhaps this is also closer to Brouwer's conception.

However, it is technically simpler (though conceptually equivalent) to separate the derivation d from the function g, so that the proof is (d,g), and also not to give the formula proved explicitly in the proof at all: this means that I shall have to say 'P is valid for A' rather than 'P is valid'. 'Validity for A' is defined as follows.

(α) An object P is <u>valid for</u> an atomic formula vacuously;

(β) P is <u>valid for</u> $A \wedge B$ iff $P = (Q,R)$ where Q is valid for A and R is valid for B;

(γ) P is <u>valid for</u> $A \vee B$ iff $P = (\text{True},Q)$ and Q is valid for A or $P = (\text{False},Q)$ and Q is valid for B.

(δ) P is <u>valid for</u> $(\exists,(\lambda x.A))$ iff $P = (n,Q)$ where Q is valid for A;

(ϵ) P is <u>valid for</u> $(\forall,(\lambda x.A))$ iff $P = (d,g)$ where d is a valid derivation tree and g is valid for '$\text{Obj} \Rightarrow A$';

(ζ) P is <u>valid for</u> $A \supset B$ iff $P = ((g,h),d)$ where d is a valid derivation tree, g is valid for '$A \Rightarrow dt$', and h is valid for '$A \Rightarrow B$';

where I am writing 'valid for $A \Rightarrow B$' to mean 'mapping any object valid for A to (if defined at all) something valid for B', 'valid for dt' to mean 'valid "as a derivation tree", ie the associated tree is well-founded', and 'valid for Obj' to mean 'any object' (ie it is vacuously true). The '$\Rightarrow$' symbol, which I introduce here for this purpose, is not to be confused with implication '$\supset$' or the sequent arrow '$\longrightarrow$'.

Thus, the above definition of an intuitionistic proof has to be rewritten: an object P is an <u>intuitionistic</u> <u>proof</u> of A iff P is valid for A and $P \vdash A$.

Let 'validity for $(\lambda x.F)t$' mean the same as 'validity for F'.

Call a *term* valid for A iff, if it is defined, it denotes a object valid for A.

Note that 'validity for A' depends only on the logical structure of A, not on what its atomic formulae are; in particular, 'validity for A' is the same as 'validity for $A(^t_x)$' for any term t.

<u>§2.2 Heyting Arithmetic</u>

I take Heyting Arithmetic (HA) to be the following system.

(U,V,t are terms; A,B,C,F,G are formulae)

Axiom schemata

(i) $A \supset (B \supset A)$

(ii) $(A \land B) \supset A$

(iii) $(A \land B) \supset B$

(iv) $A \supset (B \supset (A \land B))$

(v) $A \supset (A \lor B)$

(vi) $B \supset (A \vee B)$

(vii) $(A \vee B) \supset ((A \supset C) \supset ((B \supset C) \supset C))$

(viii) $(A \supset B) \supset ((A \supset (B \supset C)) \supset (A \supset C))$

(ix) $\mathrm{False} \supset A$

(x) $(\forall x\ A) \supset A(\frac{y}{x})$

(xi) $A(\frac{y}{x}) \supset \exists x\ A$

(xii) $At \supset \exists x\ x{=}t$ (where At is an atomic formula)

(xiii) $x = x$

(xiv) $U{=}V \supset (A(\frac{U}{x}) \supset A(\frac{V}{x}))$ (where A is an atomic formula with x not occurring within a λ-term of A)

(xv) $\mathrm{num}\ 0$

(xvi) $\mathrm{num}\ x \supset \mathrm{num}\ (Sx)$

(xvii) $Sx{=}0 \supset \mathrm{False}$

(xviii) $Sx{=}Sy \supset x{=}y$

(xix) $\mathrm{num}\ x \supset \mathrm{num}\ (fx)$, (for primitive recursive f)

(xx) $w = f(0,\underline{z}) \supset w = a\underline{z}$, $w = a\underline{z} \supset w = f(0,\underline{z})$,

$w = f(Sx,\underline{z}) \supset w = g(x,\underline{z},f(x,\underline{z}))$, $w = g(x,\underline{z},f(x,\underline{z})) \supset w = f(Sx,\underline{z})$,

(for f defined by primitive recursion from a and g)

(xxi) $A \supset B$, if A conv B (for atomic formulae A,B)

(xxii) $F \supset G$, if F conv G (for any formulae F,G)

Rules of inference

(α) $\dfrac{A \qquad A \supset B}{B}$ (β) $\dfrac{C \supset A(\frac{y}{x})}{C \supset \forall x\ A}$ (γ) $\dfrac{A(\frac{y}{x}) \supset C}{\exists x\ A \supset C}$

(where y doesn't occur free in the conclusion of (β) or (γ))

(δ) $\dfrac{F(\frac{0}{x}) \qquad \forall n\ F(\frac{n}{x}) \supset F(\frac{Sn}{x})}{\forall n\ \mathrm{num}\ n \supset F(\frac{n}{x})}$

(where x doesn't occur inside a λ-term of any atomic part of F)

Note that Axiom (xxi) is not a special case of (xxii); because of the two notations for formulae there is an ambiguity with a term A, which could represent the atomic formula A (alternatively written $\bar{A}$) or a general formula, say $(\vee,(\lambda x.(\Lambda,P,Q)))$: to resolve the ambiguity I shall always say when I intend a term to represent the atomic formula, as in (xxi), (xiv) and (xii).

My version of HA differs from the usual in that I am allowing partial functions in terms, so that various axioms have to be modified: for example, I cannot allow $(\forall x\ A) \supset A(^t_x)$ since it is not sound if t is undefined. Also, $(\lambda x.A)t$ is not always equivalent to $A(^t_x)$ (they do not convert, cf §1.5). Eg, $t \supset t$ is a theorem for undefined t whereas $(\lambda x.x \supset x)t$ is not even defined (and therefore cannot be a theorem). However, in practice this will not be an obstacle to ordinary number theory: from the above axiom system we can derive, by a straightforward structural induction on F, the theorem

$U=V \supset (F(^V_x) \supset F(^U_x))$,

and hence

$(\forall x\ F) \supset (Tt \supset F(^t_x))$,

$F(^t_x) \supset (Tt \supset \exists x\ F)$,

where U and V are terms, F is a formula with x not occurring in any λ-term of any atomic part of F, and Tt is an atomic formula (T is any term). The condition Tt implies that t is defined. In number theory all the quantifications will be relativised to numbers, and the terms will be built up from primitive recursive functions and =; so we will always be able to prove (num t) for all terms t intended to be numeric and numeric values of variables, and all formulae F will be of the form required above. This gives us $(\forall x$ num $x \supset F) \supset F(^t_x)$ and $F(^t_x) \supset (\exists x$ num $x \wedge F)$. In other words, if we define quantifiers $\forall^N$ and $\exists^N$ relativised to 'num' and restrict the language to only allow as atomic formulae $f(\underline{x})=g(\underline{y})$ for primitive recursive f and g, the usual Heyting Arithmetic axioms and rules are derivable in my system HA.

§2.3 Main theorem on Heyting Arithmetic

The following theorem shows the precise sense in which Heyting Arithmetic is justified (and hence consistent). I am not claiming that the argument is 'constructive' in any of the recognised senses; I merely claim that it is *right*.

__Theorem__ A HA derivation of a formula A (possibly containing free variables) can be transformed into a term P (possibly involving the free variables), valid for A regardless of the objects substituted for the free variables, and a valid derivation schema for $\rightarrow P \vdash A$.

__Proof__ The proof will occupy the rest of the chapter. It works by induction on the HA derivation tree. We need to find such a P and derivation for each HA axiom schema and show that the property is inherited under the HA rules of inference.

I shall proceed by reducing the problems (of finding the P's and derivations) to successively simpler problems, until obviously soluble ones are obtained. As usual, 'reducing' a problem P to a problem Q (written 'P Red. Q') means showing how to obtain a solution to P from a solution to Q. P and Q are __equivalent__ ($P \equiv Q$) iff P Red. Q and Q Red. P.

I shall use the following notation for problems:

The expression

"$F(x^X, \ldots y^Y)\{c^C, \ldots d^D\}$: $\dfrac{P \ldots Q}{R}$."

(with variables in place of x,...y,c,...d, and sequent schemata in place of P,...Q,R) denotes the problem

"Find terms x,...y (which may contain as free variables any variable occurring free in P,...Q,R), and a derivation (schema) in the sequent calculus of R from P,...Q.

The derivation must not use conversion rule (*).

The derivation must be valid, and x,...y must be valid for X,...Y, if c,...d are valid for C,...D."

(The last sentence means, for example, that in the derivation we can use

$$(\underline{x})\ DT(c_0,\ (\ \rightarrow T)) \rightarrow T$$

as an axiom if C is of the form $\forall x$ F: for c_0 is a valid derivation tree if c is valid for $\forall x$ F. The penultimate sentence is added to make Lemma 0 (§2.4) work.)

Note that I am not trying to set up a formal 'calculus of problems': my 'F'-expressions are merely abbreviated mathematical English.

As a notational simplification, if in the above problem-expression there are no premises P,...Q then write simply R instead of $\dfrac{P \ldots Q}{R}$.

Also, if there are no variables x,...y or c,...d omit the part $(x^X, \ldots y^Y)$ or $\{c^C, \ldots d^D\}$ altogether rather than writing brackets with nothing in between.

If I omit a superscript X,...Y,C,... or D it means the variable should have the same superscript as in the previous 'F'-expression. I shall use underlined metavariables to indicate lists of variables with superscripts where appropriate: in this case I shall speak of 'valid $\underline{z}$', meaning $\underline{z}$ valid for their superscripts.

§2.4 Preliminary results for HA theorems

Lemma 0 The problem $F\{\underline{Q}\}$: $(w,\underline{z})\ \Gamma \rightarrow C$ (*)

is equivalent to $F\{\underline{Q}\}$: $(\underline{z})\ \Gamma \rightarrow C$ (**)

Proof Suppose we have a solution to (**), ie a derivation of $(\underline{z})\ \Gamma \rightarrow C$; we want a solution to (*). We can obtain one by simply adding w to the list of quantified variables in each sequent schema of the given derivation (after having renamed any w variable that may occur quantified already in the derivation). The result is a valid derivation of $(w,\underline{z})\ \Gamma \rightarrow C$, ie a solution to (*).

Conversely, given a solution to (*) we can obtain a solution to (**), since we can derive $(\underline{z})\ \Gamma \rightarrow C$ from $(w,\underline{z})\ \Gamma \rightarrow C$ with the help of the Instantiation Rule.■

Lemma 1 The problem

$F(P^T)\{\underline{y}\}$: $(\underline{z})\ P_1[X_1],\ \dots\ P_k[X_k] \rightarrow P\underline{z}[X]$ (*)

reduces to

$F\{\underline{y},\underline{z}\}$: $\dfrac{X_1,\ \dots\ X_k}{X}$ (**).

Here, $X_1,\dots X_k,X$ are sequent schemata; $P_1,\dots P_k$ are terms valid for dt, ie valid trees, if $\underline{y},\underline{z}$ are valid; P occurs in (*) only where shown; T is the superscript that indicates that $P\underline{z}$ is valid for dt if $\underline{y},\underline{z}$ are valid.

Proof Suppose we have a solution of (**), ie a valid derivation D. We need a P and a derivation. Informally, P can be defined by attaching the trees $P_1,\dots P_k$ to the premises $X_1,\dots X_k$ occurring in D: the result is a derivation tree with conclusion X.

Formally, it is much easier to divide (*) into smaller problems. For each axiom A used in D we will want a solution to

(i) $F(P^T)\{\underline{y}\}$: $(\underline{z}) \longrightarrow P\underline{z}[A]$:

for each 1-premise rule $\frac{B}{A}$ used in D, and term Q, valid for dt if $\underline{y},\underline{z}$ are valid, we will want a solution to

(ii) $F(P^T)\{\underline{y}\}$: $(\underline{z})\ Q[B] \longrightarrow P\underline{z}[A]$:

and for each 2-premise rule $\frac{B\ C}{A}$ used in D, and terms Q,R, valid for dt if $\underline{y},\underline{z}$ are valid, we will want a solution to

(iii) $F(P^T)\{\underline{y}\}$: $(\underline{z})\ Q[B],\ R[C] \longrightarrow P\underline{z}[A]$.

Using solutions to these problems we can work our way up the tree D from axioms and premises to conclusion, at each sequent A having a term P and a derivation of $(\underline{z})\ P_i[X_i],\ \ldots\ P_j[X_j] \longrightarrow P\underline{z}[A]$ where $X_i,\ldots X_j$ are the premises used in deriving A. When we get to the conclusion we have a solution to (*).

It remains only to solve Problems (i),(ii) and (iii). (i) is solved by putting $P\underline{z} \equiv (A,nil)$: A is a legitimate axiom if $\underline{y},\underline{z}$ are valid, by assumption, so P is valid for T; and $P\underline{z}$ is a derivation schema for A, so $P\underline{z}[A]$ converts to True; so the sequent in Problem (i) is validly derivable.

Problem (ii): define $P\underline{z} \equiv (A,(Q,nil))$. Consider the following derivations:

$$\begin{cases} (\underline{z})\ Q[B] \longrightarrow (\text{deriv } Q)\ \&\ (\pi_0 Q = B), \\ (\underline{z})\ (\text{deriv } Q)\ \&\ (\pi_0 Q = B) \longrightarrow (\text{deriv } (P\underline{z}))\ \&\ (\pi_0(P\underline{z}) = A), \\ (\underline{z})\ (\text{deriv } (P\underline{z}))\ \&\ (\pi_0(P\underline{z}) = A) \longrightarrow P\underline{z}[A]. \end{cases}$$

Combined using Cut, they yield the desired sequent. The first and last follow immediately from definitions; the second may be derived by Instantiation from

$(x,\underline{z})\ (\text{deriv } x)\ \&\ (\pi_0 x = B) \longrightarrow \text{deriv}(A,(x,nil))\ \&\ (\pi_0(A,(x,nil)) = A)$.

But $\pi_0(A,(x,nil))$ conv A, so the RHS becomes simply $\text{deriv}(A,(x,nil))$, which by the defining equation becomes $\mathrm{inf}_1(\pi_0 x,A)$ & (deriv x). Hence we merely have to derive

$(x,\underline{z})$ (deriv x) & $(\pi_0 x = B) \rightarrow \mathrm{inf}_1(\pi_0 x,A)$ & (deriv x),

which is easy since we can replace $\pi_0 x$ by B in the RHS, and $\mathrm{inf}_1(B,A)$ converts to True. If $\underline{y},\underline{z}$ are valid then Q is a valid tree, by assumption, so $P\underline{z}$ is also, as required.

Problem (iii) is similar to (ii), but with two premises Q[B], R[C] instead of one.∎

<u>Lemma 2</u> We can solve $F(P^T)$: $(\underline{x})\ X \rightarrow P\underline{x}[\ \rightarrow X]$

where X is a term and T is the superscript that indicates that $P\underline{x}$ is valid for dt for all $\underline{x}$.

<u>Proof</u> Define P by

$P\underline{x} \equiv$ the derivation tree $((\ \rightarrow X),(((\ \rightarrow \mathrm{True}),\mathrm{nil}),((\mathrm{True} \rightarrow X),\mathrm{nil})))$,

which consists of $\rightarrow X$ inferred in one step from $\rightarrow$ True and True $\rightarrow X$. To derive $(\underline{x})\ X \rightarrow P\underline{x}[\ \rightarrow X]$, first replace the antecedent X by X=True; then 'evaluate' $P\underline{x}[\ \rightarrow X]$, ie convert it to True by following the computation of $i(P\underline{x}[\ \rightarrow X])$: the single inference $\dfrac{\rightarrow \mathrm{True} \qquad \mathrm{True} \rightarrow X}{\rightarrow X}$ is validated as an application of Cut, $\rightarrow$ True is recognised as an axiom, and True $\rightarrow$ X is recognised as an axiom with the help of Conversion Rule (*) and the hypothesis X=True. And of course X=True $\rightarrow$ True is derivable.∎

<u>Lemma 3</u> We can solve

$$F(\underline{Q}):\quad (\underline{z},\underline{y})\ A\begin{bmatrix}\underline{y}\\ \underline{x}\end{bmatrix},\ \ldots\ B\begin{bmatrix}\underline{y}\\ \underline{x}\end{bmatrix},\ P[(\underline{x})\ A,\ldots B \rightarrow C] \rightarrow C\begin{bmatrix}\underline{y}\\ \underline{x}\end{bmatrix}$$

where P is a term valid for dt for all $\underline{y},\underline{z}$ and valid $\underline{Q}$.

__Proof__ The problem reduces, by Lemma 0, to

$$F\{\underline{Q}\}:\quad A\begin{bmatrix}\underline{y}\\ x\end{bmatrix},\ \ldots\ B\begin{bmatrix}\underline{y}\\ x\end{bmatrix},\ P[(\underline{x})\ A,\ldots B \longrightarrow C] \longrightarrow C\begin{bmatrix}\underline{y}\\ x\end{bmatrix},$$

which reduces, by Cut, to

$$\begin{cases} F(U^{d*}):\quad A\begin{bmatrix}\underline{y}\\ x\end{bmatrix} \longrightarrow U[\ \longrightarrow A\begin{bmatrix}\underline{y}\\ x\end{bmatrix}] \\ \vdots \\ F(V^{d*}):\quad B\begin{bmatrix}\underline{y}\\ x\end{bmatrix} \longrightarrow V[\ \longrightarrow B\begin{bmatrix}\underline{y}\\ x\end{bmatrix}] \\ F(W^{d*})\{\underline{Q}\}: U[\ \longrightarrow A\begin{bmatrix}\underline{y}\\ x\end{bmatrix}],\ldots V[\ \longrightarrow B\begin{bmatrix}\underline{y}\\ x\end{bmatrix}],\ P[(\underline{x})\ A,\ldots B \longrightarrow C] \longrightarrow W[\ \longrightarrow C\begin{bmatrix}\underline{y}\\ x\end{bmatrix}] \\ F\{\underline{Q}\}:\quad W[\ \longrightarrow C\begin{bmatrix}\underline{y}\\ x\end{bmatrix}] \longrightarrow C\begin{bmatrix}\underline{y}\\ x\end{bmatrix} \end{cases}$$

(When I write 'W' in the last problem I mean of course that W is the solution found for the previous problem; likewise U,...V in the penultimate problem mean the solutions found to the previous problems; I shall always follow this convention when problems are bracketed together like this.)

Now, the first problems are solved by Lemma 2; the last is a valid reflection principle; that only leaves the penultimate problem, which reduces, by Lemma 1 to

$$F\{\underline{Q}\}:\quad \frac{\longrightarrow A\begin{bmatrix}\underline{y}\\ x\end{bmatrix}\quad \ldots\quad \longrightarrow B\begin{bmatrix}\underline{y}\\ x\end{bmatrix}\qquad (\underline{x})\ A,\ldots B \longrightarrow C}{\longrightarrow C\begin{bmatrix}\underline{y}\\ x\end{bmatrix}},$$

which is easy.∎

One difficulty with finding terms T such that $T \vdash A \supset B$ is that $((T_b,T_c),T_a) \vdash A \supset B$ does not convert to $T_a[(Q) \longrightarrow T_bQ[Q \vdash A \longrightarrow T_cQ \vdash B]]$, as one might hope, because of the restrictions on λ-conversion (§1.5): the best one can do is

$$((T_b,T_c),T_a) \vdash A \supset B \quad \text{conv} \quad (\lambda wuv.\ ((u,v),w) \vdash A \supset B)T_aT_bT_c$$

conv $(\lambda wuv.\ w[(Q) \longrightarrow uQ[Q \vdash A \longrightarrow vQ \vdash B]])T_a T_b T_c$

conv $(\lambda uv.\ T_a[(Q) \longrightarrow uQ[Q \vdash A \longrightarrow vQ \vdash B]])T_b T_c$.

A similar problem arises with universal formulae. The next lemma is an attempt to handle such expressions.

<u>Lemma 4</u> $F(P^T)\{\underline{R}\}$: $(\underline{Q})\ \Gamma \longrightarrow (\lambda\underline{u}.\ P\underline{Q}[(\underline{v})\ A \longrightarrow C])\underline{X}$ (*)

$$\text{Red.}\begin{cases} F\{\underline{Q},\underline{R}\}:\ (\underline{v})\ \Gamma,\ A \longrightarrow (\lambda\underline{u}.C)\underline{X} & (**) \\ F\{\underline{R}\}:\ \Gamma \longrightarrow \overline{\text{True}}\ (\underline{X}) & (***) \end{cases}$$

(where P occurs only where shown; $\underline{Q}$ do not occur in Γ or $\underline{X}$; $\underline{v},\underline{u}$ occur only in C; T is the superscript that indicates that $P\underline{Q}$ is valid for dt if $\underline{Q}$ are valid).

<u>Proof</u> Problem (*) reduces to

$$\begin{cases} F(P^T)\{\underline{R}\}:\ (\underline{u},\underline{Q})\ \Gamma,\ \underline{u}=\underline{X} \longrightarrow P\underline{Q}[(\underline{v})\ A \longrightarrow C] & (1) \\ F\{\underline{R}\}:\ \Gamma \longrightarrow \overline{\text{True}}\ (\underline{X}) \quad \text{ie, } (***) \end{cases}$$

(where $\underline{X}$ is $X_1,\ldots X_n$, $\underline{u}$ is $u_1,\ldots u_n$, $\underline{u}=\underline{X}$ is short for $u_1=X_1,\ldots,u_n=X_n$). For we can convert $P\underline{Q}[(\underline{v})\ A \longrightarrow C]$ to $(\lambda\underline{u}.\ P\underline{Q}[(\underline{v})\ A \longrightarrow C])\underline{u}$ in (1), then instantiate $\underline{u}$ to $\underline{X}$, then convert $\underline{X}=\underline{X}$ in the antecedent to $\overline{\text{True}}\ (\underline{X})$, then apply Cut with (***) (with redundant variables $\underline{Q}$ added) to get (*). It only remains to reduce (1) to (**).

Now, defining P by $P\underline{Q} \equiv P'\underline{X}\underline{Q}$, (1) clearly reduces to

$F(P'^{\sigma})\{\underline{R}\}$: $(\underline{u},\underline{Q})\ \Gamma,\ \underline{u}=\underline{X} \longrightarrow P'\underline{u}\underline{Q}[(\underline{v})\ A \longrightarrow C]$

(σ is the superscript that indicates that $P'\underline{u}\underline{Q}$ is valid for dt if $\underline{Q}$ are valid); and (writing $\Gamma \equiv A_1,\ldots A_k$) this reduces, by Cut, to

$$\begin{cases} F(P^{?}_{(1)}):\ (\underline{u},\underline{Q})\ A_1 \ \longrightarrow\ P_{(1)}\underline{uQ}[\ \longrightarrow A_1] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ F(P^{?}_{(k)}):\ (\underline{u},\underline{Q})\ A_k \ \longrightarrow\ P_{(k)}\underline{uQ}[\ \longrightarrow A_k] \\ F(T^{?}_{(1)}):\ (\underline{u},\underline{Q})\ u_1{=}X_1 \ \longrightarrow\ T_{(1)}\underline{uQ}[\ \longrightarrow u_1{=}X_1] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ F(T^{?}_{(n)}):\ (\underline{u},\underline{Q})\ u_n{=}X_n \ \longrightarrow\ T_{(n)}\underline{uQ}[\ \longrightarrow u_n{=}X_n] \\ F(P')\{\underline{R}\}:\ (\underline{u},\underline{Q})\ P_{(1)}\underline{uQ}[\ \longrightarrow A_1],\ \ldots\ P_{(k)}\underline{uQ}[\ \longrightarrow A_k],\ T_{(1)}\underline{uQ}[\ \longrightarrow u_1{=}X_1], \\ \qquad\qquad \ldots\ T_{(n)}\underline{uQ}[\ \longrightarrow u_n{=}X_n]\ \longrightarrow\ P'\underline{uQ}[(\underline{y})\ \Delta \longrightarrow C] \end{cases}$$

all of which are solved by Lemma 2 except for the last, which reduces, by Lemma 1, to

$$F\{\underline{Q},\underline{R}\}:\quad \frac{\longrightarrow A_1 \quad \ldots \quad \longrightarrow A_k \quad \longrightarrow u_1{=}X_1 \quad \ldots \quad \longrightarrow u_n{=}X_n}{(\underline{y})\ \Delta \longrightarrow C};$$

which can be rearranged to

$F\{\underline{Q},\underline{R}\}$: $(\underline{y})\ \Gamma,\ \underline{u}{=}\underline{X},\ \Delta \longrightarrow C$;

then convert C to $(\lambda\underline{u}.C)\underline{u}$, replace that by $(\lambda\underline{u}.C)\underline{X}$, and apply Weakening, so that our problem reduces to

$F\{\underline{Q},\underline{R}\}$: $(\underline{y})\ \Gamma,\ \Delta \longrightarrow (\lambda\underline{u}.C)\underline{X}$, ie (**), as required.∎

The next lemma may be regarded as justifying the claim made at the end of §2.0 that the '(∀,f)' clause in the definition of ⊢ encompasses all ways of using protologic to establish that (a function of x) ⊢ fx for any x.

Lemma 5 $\qquad F(T^{\forall x\ A})\{\underline{R}\}$: $\Gamma \longrightarrow T \vdash \forall x\ A \qquad$ (*)

is equivalent to $F(T'^{A})\{\underline{R}\}$: $\Gamma \longrightarrow T' \vdash A(\tfrac{y}{x}) \qquad$ (**)

where y does not occur free in (*). If the condition on y is dropped then (**) still Red. (*).

Proof Suppose we have a solution to (**) and y doesn't occur free in (*). Choose a new variable z, not occurring in (*) or (**), and substitute it for free occurrences of x in (*): the problem becomes

$F(T^{\forall x\ A})\{\underline{R}\}$: $\Gamma(\tfrac{z}{x}) \longrightarrow T \vdash \forall x\ A$.

Now define $T \equiv (T_a,T_b)$, where $T_b \equiv (\lambda y.T'\binom{z}{x})$, T' being taken from the given solution to (***); our problem reduces to

$F(T_a^{d*})\{\underline{R}\}$: $\Gamma\binom{z}{x} \longrightarrow (\lambda u.\ T_a[(x) \longrightarrow ux \vdash A])T_b$,

Red. (Lemma 4) $\begin{cases} F\{\underline{R}\}\colon\ (x)\ \Gamma\binom{z}{x} \longrightarrow \underline{(\lambda u.}\ ux \vdash A)T_b \\ F\{\underline{\bar{R}}\}\colon\qquad \Gamma\binom{z}{x} \longrightarrow \text{True}\ T_b \end{cases}$

of which the second is trivial since T_b is a λ-term, and therefore Manifestly Well-Defined; and the first reduces to

$F\{\underline{R}\}$: $(x)\ \Gamma\binom{z}{x} \longrightarrow T_b x \vdash A$

Red. (Lemma 0) $F\{\underline{R}\}$: $\Gamma\binom{z}{x} \longrightarrow T'\binom{z}{x}\binom{x}{y} \vdash A$;

substituting y for x, then x for z, gives (***), as required.

Conversely, given (*), (***) reduces, by Cut, to

$\begin{cases} F(T^{\forall x\ A})\{\underline{R}\}\colon\ \ \Gamma \longrightarrow T \vdash \forall x\ A \\ F(T'^{A})\{\underline{R}\}\colon\ \ T \vdash \forall x\ A \ \longrightarrow\ T' \vdash A\binom{y}{x} \end{cases}$

the first of which is simply (*). The other reduces to

$F(T')\{\underline{R}\}$: $(u)\ u{=}T_1,\ T_0[(x) \longrightarrow ux \vdash A] \longrightarrow T' \vdash A\binom{y}{x}$

(recall that T_0 is $\pi_0 T$, T_1 is $\pi_1 T$ (§2.0)); this is verified by instantiating u to T_1. The problem reduces, by Cut, to

$\begin{cases} F\{\underline{R}\}\colon\ \ (u)\ T_0[(x) \longrightarrow ux \vdash A] \longrightarrow uy \vdash A\binom{y}{x} \\ F(\bar{T}')\{\underline{R}\}\colon\ \ (u)\ u{=}T_1,\ uy \vdash A\binom{y}{x} \longrightarrow T' \vdash A\binom{y}{x} \end{cases}$

the first of which is solved by Lemma 0 and Lemma 3; the second is solved by setting $T' \equiv T_1 y$.∎

<u>Lemma 6</u> We can solve $F\{\underline{R}\}$: $(\underline{x})\ U \vdash A,\ V \vdash A \supset B \longrightarrow V_3 U \vdash B$ (where U and V are valid for A and $A \supset B$ if $\underline{R}$ are valid).

<u>Proof</u> By the sequent calculus, the problem reduces to

$F\{\underline{R}\}$: $(\underline{x},p,q,r)$ $p{=}U$, $p \vdash A$, $q{=}V_2$, $r{=}V_3$,

$$V_1[(Q) \longrightarrow qQ[Q \vdash A \longrightarrow rQ \vdash B]] \ \longrightarrow\ V_3 U \vdash B,$$

which reduces, by Cut, to

$$\begin{cases} F\{\underline{R}\}: & (\underline{x},p,q,r)\ V_1[(Q) \longrightarrow qQ[Q \vdash A \longrightarrow rQ \vdash B]] \longrightarrow qp[p \vdash A \longrightarrow rp \vdash B] \\ F\{\overline{\underline{R}}\}: & (\underline{x},p,q,r)\ p{=}U,\ q{=}V_2,\ qp[p \vdash A \longrightarrow rp \vdash B] \longrightarrow V_2U[p \vdash A \longrightarrow rp \vdash B] \\ F\{\overline{\underline{R}}\}: & (\underline{x},p,q,r)\ p \vdash A,\ V_2U[p \vdash A \longrightarrow rp \vdash B] \longrightarrow rp \vdash B \\ F\{\overline{\underline{R}}\}: & (\underline{x},p,q,r)\ p{=}U,\ r{=}V_3,\ rp \vdash B \longrightarrow V_3U \vdash B \end{cases}$$

which are all solved either immediately or by Lemma 3.■

The next lemma justifies the claim made at the end of §2.0 that the '$A \supset B$' clause in the definition of $\vdash$ encompasses all ways of protologically establishing that if $Q \vdash A$ and Q is valid for A then (a function of Q) $\vdash B$.

<u>Lemma 7</u> $\quad F(T^{A\supset B})\{\underline{R}\}:\ \Gamma \longrightarrow T \vdash A \supset B \qquad (*)$

is equivalent to $\quad F(T'^{B})\{Q^{A},\underline{R}\}:\ \Gamma,\ Q \vdash A \longrightarrow T' \vdash B \qquad (**)$.

<u>Proof</u> Suppose we have a solution to ($**$); using the T' given define $T_c \equiv (\lambda Q.T')$ (which is valid for $A \Rightarrow B$). We can solve both of

$$\begin{cases} F\{Q^{A},\underline{R}\}: & \Gamma,\ \underline{Q \vdash} A \longrightarrow (\lambda v.\ vQ \vdash B)T_c \\ F\{\underline{R}\}: & \Gamma \longrightarrow \overline{\text{True}}\ T_c \end{cases}$$

whence, by Lemma 4, we get a solution to

$F(T_b^{A\Rightarrow dt})\{\underline{R}\}:\ \ (Q)\ \Gamma \longrightarrow (\lambda v.\ T_bQ[Q \vdash A \longrightarrow vQ \vdash B]])T_c$.

Using the T_b so obtained, we can solve both of

$$\begin{cases} F\{\underline{R}\}: & (Q)\ \ \Gamma\ \underline{\longrightarrow}\ (\lambda uv.\ uQ[Q \vdash A \longrightarrow vQ \vdash B]])T_bT_c \\ F\{\overline{\underline{R}}\}: & \Gamma \longrightarrow \overline{\text{True}}\ (T_b,T_c) \end{cases}$$

whence, by Lemma 4 again, we obtain a solution to

$F(T_a^{d*})\{\underline{R}\}:\ \ \Gamma \longrightarrow (\lambda uv.\ T_a[(Q) \longrightarrow uQ[Q \vdash A \longrightarrow vQ \vdash B]])T_bT_c$.

Now, the right-hand side converts to $((T_b,T_c),T_a) \vdash A \supset B$; and so, defining $T \equiv ((T_b,T_c),T_a)$, we get a solution to

$F(T^{A\supset B})\{\underline{R}\}:\ \ \Gamma \longrightarrow T \vdash A \supset B,\qquad$ ie ($*$), as required.

Conversely, (**) reduces, defining $T' \equiv T_3 Q$ and using Cut, to

$$\begin{cases} F(T^{A\supset B})\{\underline{R}\}: & \Gamma \longrightarrow T \vdash A \supset B \\ F\{Q,\underline{R}\}: & Q \vdash A,\ T \vdash A \supset B \longrightarrow T_3 Q \vdash B \end{cases}$$

the first is (*), and the second is solved by Lemma 6.■

§2.5 The axioms of Heyting Arithmetic proved

The problem is, for each axiom schema A, to solve $F(T^A)$: $\longrightarrow T \vdash A$.

Axiom (i): $A \supset (B \supset A)$

The problem is $F(T^{A\supset(B\supset A)})$: $\longrightarrow T \vdash A \supset (B \supset A)$,

$\equiv$ (by Lemma 7) $F(T'^{B\supset A})\{Q^A\}$: $Q \vdash A \longrightarrow T' \vdash B \supset A$,

$\equiv$ (by Lemma 7) $F(T''^A)\{Q^A,R^B\}$: $Q \vdash A,\ R \vdash B \longrightarrow T'' \vdash A$,

which is solved by defining T'' by $T'' \equiv Q$.

Axiom (ii): $(A \wedge B) \supset A$

The problem is $F(T^{(A\wedge B)\supset A})$: $\longrightarrow T \vdash (A \wedge B) \supset A$,

$\equiv$ (Lemma 7) $F(T'^A)\{Q^{A\wedge B}\}$: $Q \vdash A \wedge B \longrightarrow T' \vdash A$,

which is solved by defining $T' \equiv Q_0$, since $Q \vdash A \wedge B$ conv $Q_0 \vdash A$ & $Q_1 \vdash B$, and X&Y $\longrightarrow$ X is derivable.

Axiom (iii): $(A \wedge B) \supset B$

This is very similar to the case of Axiom (ii).

Axiom (iv): $(A \supset (B \supset (A \wedge B))$

The problem is $F(T^{A\supset(B\supset(A\wedge B))})$: $\longrightarrow T \vdash (A \supset (B \supset (A \wedge B))$,

$\equiv$ (Lemma 7 twice) $F(T''^{A\wedge B})\{Q^A,R^B\}$: $Q \vdash A,\ R \vdash B \longrightarrow T'' \vdash A \wedge B$,

which is solved by setting $T'' \equiv (Q,R)$, since $(Q,R) \vdash A \wedge B$ conv $Q \vdash A$ & $R \vdash B$, and X, Y $\longrightarrow$ X&Y is derivable.

Axiom (v): $A \supset (A \vee B)$

The problem is $F(T^{A \supset (A \vee B)})$: $\longrightarrow T \vdash A \supset (A \vee B)$,

$\equiv$ (Lemma 7) $F(T'^{A \vee B})\{Q^{A}\}$: $Q \vdash A \longrightarrow T' \vdash A \vee B$,

which is solved by setting $T' \equiv (\text{True}, Q)$, since $(\text{True}, Q) \vdash A \vee B$ conv $Q \vdash A$.

Axiom (vi): $B \supset (A \vee B)$

This is very similar to the case of Axiom (v).

Axiom (vii): $(A \vee B) \supset ((A \supset C) \supset ((B \supset C) \supset C))$

The problem is

$F(T^{(A \vee B) \supset ((A \supset C) \supset ((B \supset C) \supset C))})$: $\longrightarrow T \vdash (A \vee B) \supset ((A \supset C) \supset ((B \supset C) \supset C))$,

$\equiv$ (Lemma 7 three times)

$F(T'''^{C})\{Q^{A \vee B}, R^{A \supset C}, U^{B \supset C}\}$: $Q \vdash A \vee B$, $R \vdash A \supset C$, $U \vdash B \supset C \longrightarrow T''' \vdash C$.

Now, $Q \vdash A \vee B$ conv (if Q_0 then $Q_1 \vdash A$ else $Q_1 \vdash B$). Define T''' by $T''' \equiv$ if Q_0 then R_3Q_1 else U_3Q_1; also define $X \equiv$ if Q_0 then R_2Q_1 else U_2Q_1: these are valid for C and dt respectively if Q, R and U are valid for $A \vee B$, $A \supset C$ and $B \supset C$. By the If Rule our problem reduces to the two problems

$$\begin{cases} F\{Q,R,U\}: & Q_0,\ Q_1 \vdash A,\ R \vdash A \supset C \longrightarrow R_3Q_1 \vdash C, \\ F\{Q,R,U\}: & \text{not } Q_0,\ Q_1 \vdash B,\ U \vdash B \supset C \longrightarrow U_3Q_1 \vdash C. \end{cases}$$

I shall only solve the first problem: the other is very similar. We cannot just apply Lemma 6 because Q_1 is not necessarily valid for A (only if in fact Q_0 = True); but we can argue similarly to the proof of Lemma 6, as follows. The problem reduces to

$F\{Q,R,U\}$: (p,q,r) Q_0, $r = Q_1$, $r \vdash A$, $p = R_2$,

$$R_1[(u) \longrightarrow pu[u \vdash A \longrightarrow qu \vdash C]] \longrightarrow qr \vdash C,$$

by instantiating r,p,q to Q_1,R_2,R_3. By Cut this reduces to

$$\begin{cases} F\{Q,R,U\}:\ (p,q,r)\ R_1[(u) \to pu[u \vdash A \to qu \vdash C]] \to pr[r \vdash A \to qr \vdash C] \\ F\{Q,R,U\}:\ (p,q,r)\ Q_0,\quad r{=}Q_1,\quad p{=}R_2,\quad pr[r \vdash A \to qr \vdash C] \to \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad X[r \vdash A \to qr \vdash C] \\ F\{Q,R,U\}:\ (p,q,r)\ r \vdash A,\quad X[r \vdash A \to qr \vdash C] \to qr \vdash C \end{cases}$$

the first and third problems are solved by Lemma 3; the second is easy, using the definition of X.

Axiom (viii): $(A \supset B) \supset ((A \supset (B \supset C)) \supset (A \supset C))$

The problem is

$F(T^{(A\supset B)\supset((A\supset(B\supset C))\supset(A\supset C))}):\ \to T \vdash (A \supset B) \supset ((A \supset (B \supset C)) \supset (A \supset C)),$

$\equiv$ (Lemma 7 three times) $F(T''^{C})\{Q^{A\supset B}, R^{A\supset(B\supset C)}, U^{A}\}:$

$$Q \vdash A \supset B,\quad R \vdash A \supset (B \supset C),\quad U \vdash A \ \to\ T'' \vdash C,$$

which, defining T'' by $T'' \equiv (R_3U)_3(Q_3U)$, reduces to the three problems

$$\begin{cases} F\{Q,R,U\}:\ Q \vdash A \supset B,\quad U \vdash A \to Q_3U \vdash B, \\ F\{Q,R,U\}:\ R \vdash A \supset (B \supset C),\quad U \vdash A \to R_3U \vdash B \supset C, \\ F\{Q,R,U\}:\ Q_3U \vdash B,\quad R_3U \vdash B \supset C \to (R_3U)_3(Q_3U) \vdash C. \end{cases}$$

All three problems are solved by Lemma 6.

Axiom (ix): False $\supset$ A

The problem is $F(T^{False\supset A}):\ \to T \vdash \text{False} \supset A,$

$\equiv$ (Lemma 7) $F(T'^{A})\{Q^{False}\}:\ Q \vdash \text{False} \to T' \vdash A.$

But $Q \vdash$ False conv False, and False $\to X$ is derivable, for any X.

So we can choose $T' \equiv$ a fixed object valid for A.

Axiom (x): $(\forall x\ A) \supset A(\frac{y}{x})$

The problem is $F(T^{(\forall x\ A)\supset A(\frac{y}{x})}):\ \to T \vdash (\forall x\ A) \supset A(\tfrac{y}{x}),$

$\equiv$ (Lemma 7) $F(T'^{A})\{Q^{\forall x\ A}\}:\ Q \vdash \forall x\ A \to T' \vdash A(\tfrac{y}{x}),$

Red. (Lemma 5) $F(T''^{\forall x\ A})\{Q^{\forall x\ A}\}:\ Q \vdash \forall x\ A \to T'' \vdash \forall x\ A,$

which is solved by defining $T'' \equiv Q$.

Axiom (xi): $A(\substack{y\\x}) \supset \exists x\, A$

Write the axiom as $fy \supset (\exists,f)$, where $f \equiv (\lambda x.A)$, using Axiom (xxii). The problem is $F(T^{fy\supset(\exists,f)})$: $\longrightarrow$ $T \vdash fy \supset (\exists,f)$,

$\equiv$ (Lemma 7) $F(T'^{(\exists,f)})\{Q^{fy}\}$: $Q \vdash fy$ $\longrightarrow$ $T' \vdash (\exists,f)$.

But $Q \vdash fy$ conv $(y,Q) \vdash (\exists,f)$, by the definition of $\vdash$, so we simply define $T' \equiv (y,Q)$.

Axiom (xii) $At \supset \exists x\ x=t$ (At atomic)

The problem is $F(T^{At\supset\exists x\ x=t})$: $\longrightarrow$ $T \vdash At \supset \exists x\ x=t$,

$\equiv$ (Lemma 7) $F(T'^{\exists x\ x=t})\{Q^{At}\}$: $Q \vdash At$ $\longrightarrow$ $T' \vdash \exists x\ x=t$;

define $T' \equiv (t,0)$, so that our problem is to derive

$$At \longrightarrow t=t,$$

which is trivial.

Axiom (xiii): $x = x$

Axiom (xv): num 0

These both convert to True, so any object will do as a proof.

Axiom (xiv): $U=V \supset (A(\substack{u\\x}) \supset A(\substack{v\\x}))$ (A is atomic with x not occurring within a λ-term)

The problem is $F(T^{U=V\supset(A\supset A)})$: $\longrightarrow$ $T \vdash U=V \supset (A(\substack{u\\x}) \supset A(\substack{v\\x}))$,

$\equiv$ (Lemma 7) $F(T'^{A})\{Q^{U=V},R^{A}\}$: $Q \vdash U=V$, $R \vdash A(\substack{u\\x})$ $\longrightarrow$ $T' \vdash A(\substack{v\\x})$.

Of course, $Q \vdash U=V$ conv $U=V$; so, defining $T' \equiv R$, we merely have to derive

$$U=V,\ R \vdash A(\substack{u\\x}) \longrightarrow R \vdash A(\substack{v\\x}),$$

which is an axiom.

Axiom (xvi): num x $\supset$ num (Sx)

Axiom (xvii): Sx=0 $\supset$ False

Axiom (xviii): Sx=Sy $\supset$ x=y

Axiom (xix): num x $\supset$ num (fx), for primitive recursive f

Axiom (xx): $w{=}f(0,\underline{z}) \supset w{=}a\underline{z}$, $\quad w{=}a\underline{z} \supset w{=}f(0,\underline{z})$,

$$w{=}f(Sx,\underline{z}) \supset w{=}g(x,\underline{z},f(x,\underline{z})), \qquad w{=}g(x,\underline{z},f(x,\underline{z})) \supset w{=}f(Sx,\underline{z})$$

(for f defined by primitive recursion from a and g)

Axiom (xxi): A $\supset$ B, for atomic formulae A,B such that A conv B

All these are of the form A $\supset$ B, with A and B atomic. The problem is $F(T^{A\supset B})$: $\longrightarrow$ T $\vdash$ A $\supset$ B, $\equiv$ (Lemma 7) $F(T'^{B})\{Q^{A}\}$: Q $\vdash$ A $\longrightarrow$ T' $\vdash$ B. But of course Q $\vdash$ A conv A and T' $\vdash$ B conv B , so, defining T' to be, say, Q (it will be automatically valid for B), our problem is to derive the sequent A $\longrightarrow$ B, which in each case can be done (see §1.6).

Axiom (xxii): F $\supset$ G, for formulae F,G such that F conv G

The problem is $F(T^{F\supset G})$: $\longrightarrow$ T $\vdash$ F $\supset$ G, which by Lemma 7 reduces to $F(T'^{G})\{Q^{F}\}$: Q $\vdash$ F $\longrightarrow$ T' $\vdash$ G,
which is solved by defining T' $\equiv$ Q since F conv G.

§2.6 The rules of Heyting Arithmetic justified

The problem is, for each rule $\frac{A \ \ldots \ B}{C}$, to reduce $F(T^{C})$: $\longrightarrow$ T $\vdash$ C to $F(T'^{A})$: $\longrightarrow$ T' $\vdash$ A, ... $F(T''^{B})$: $\longrightarrow$ T'' $\vdash$ B.

Rule (α): $\dfrac{A \qquad A \supset B}{B}$

$F(T^{B})$: $\longrightarrow T \vdash B$ Red. (by Cut)

$$\begin{cases} F(T'^{A}): & \longrightarrow\ T' \vdash A, \\ F(T''^{A\supset B}): & \longrightarrow\ T'' \vdash A \supset B, \\ F: & T' \vdash A,\quad T'' \vdash A \supset B \ \longrightarrow\ T''{}_{3}T' \vdash B, \end{cases}$$

defining $T \equiv T''{}_{3}T'$: the first two are given and the third is solved by Lemma 6.

Rule (β): $\dfrac{C \supset A(\frac{y}{x})}{C \supset \forall x\ A}$ (y not occurring free in the conclusion)

$F(T^{C\supset(\forall x\ A)})$: $\longrightarrow\ T \vdash C \supset \forall x\ A$,

$\equiv$ (Lemma 7) $F(T'^{\forall x\ A})\{Q^{C}\}$: $Q \vdash C \ \longrightarrow\ T' \vdash \forall x\ A$,

$\equiv$ (Lemma 5) $F(T''^{A})\{Q^{C}\}$: $Q \vdash C \ \longrightarrow\ T'' \vdash A(\frac{y}{x})$,

$\equiv$ (Lemma 7) $F(T'''^{C\supset A})$: $\longrightarrow\ T''' \vdash C \supset A(\frac{y}{x})$, as required.

Rule (γ): $\dfrac{A(\frac{y}{x}) \supset C}{(\exists x\ A) \supset C}$ (y not occurring free in the conclusion)

Assume $F(P^{A(\frac{y}{x})\supset C})$: $\longrightarrow\ P \vdash A(\frac{y}{x}) \supset C$ (*) is solved; we need to solve $F(T^{(\exists x\ A)\supset C})$: $\longrightarrow\ T \vdash (\exists x\ A) \supset C$,

$\equiv$ (Lemma 7) $F(T'^{C})\{Q^{\exists x\ A}\}$: $Q \vdash \exists x\ A \ \longrightarrow\ T' \vdash C$,

Red. (defining $T' \equiv ((\lambda y.P)Q_0)_3 Q_1$) $F\{Q\}$: (y) $y{=}Q_0$, $Q_1 \vdash A(\frac{y}{x}) \longrightarrow T' \vdash C$,

$$\text{Red.}\begin{cases} F: & (y)\ \longrightarrow\ P \vdash A(\frac{y}{x}) \supset C, \\ F\{Q\}: & (y)\ y{=}Q_0,\ Q_1 \vdash A(\frac{y}{x}),\ P \vdash A(\frac{y}{x}) \supset C \longrightarrow T' \vdash C; \end{cases}$$

the first reduces by Lemma 0 to (*); the second reduces to

$F\{Q\}$: (y) $Q_1 \vdash A(\frac{y}{x})$, $(\lambda y.P)Q_0 \vdash A(\frac{y}{x}) \supset C \longrightarrow T' \vdash C$,

which is solved by Lemma 6.

Rule (δ): $\dfrac{F(\frac{0}{x}) \qquad \forall n\ F(\frac{n}{x}) \supset F(\frac{Sn}{x})}{\forall n\ \mathrm{num}\ n \supset F(\frac{n}{x})}$

First note that it is sufficient to justify $\dfrac{f0 \qquad \forall n\ fn \supset f(Sn)}{\forall n\ \mathrm{num}\ n \supset fn}$, where $f \equiv (\lambda x.F)$, since $f0$ conv $F(\frac{0}{x})$, fn conv $F(\frac{n}{x})$, and $F(\frac{Sn}{x}) \supset f(Sn)$ is a theorem of HA\Rule δ (using the theorem $U{=}V \supset (F({}^{U}_{x}) \supset F({}^{V}_{x}))$ of §2.2).

We are given solutions to

$F(a^{F})$: $\longrightarrow$ $a \vdash f0$ and $F(T^{\forall n\; F\supset F})$: $\longrightarrow$ $T \vdash \forall n\; fn \supset f(Sn)$;

the latter $\equiv$ (by Lemma 5) $F(T'^{F\supset F})$: $\longrightarrow$ $T' \vdash fm \supset f(Sm)$ (*).

Note that I can write as superscripts F and $\forall n\; F\supset F$ instead of $f0$ and $\forall n\; fn \supset f(Sn)$, since, as remarked at the end of §2.1, validity only depends on the logical structure of the formula.

The problem to be solved is $F(P^{\forall n\; \mathrm{num}\; n \supset F})$: $\longrightarrow$ $P \vdash \forall n\; \mathrm{num}\; n \supset fn$,

$\equiv$ (Lemma 5) $F(P'^{\mathrm{num}\; m \supset F})$: $\longrightarrow$ $P' \vdash \mathrm{num}\; m \supset fm$,

$\equiv$ (Lemma 7) $F(P''^{F})(Q^{\mathrm{num}\; m})$: $Q \vdash \mathrm{num}\; m$ $\longrightarrow$ $P'' \vdash fm$,

Red. (Lemma 0) F: (m) $\mathrm{num}\; m$ $\longrightarrow$ $\mathcal{F}m \vdash fm$,

defining $P'' \equiv \mathcal{F}m$, where $\mathcal{F}$ is the function defined by primitive recursion from a and g, where a is given above and $gm \equiv T'_3$, where T' is given in (*). Our problem reduces, using the protological Induction Rule (§1.6):

$$\begin{cases} F\colon \longrightarrow \mathcal{F}0 \vdash f0, \\ F\colon (m)\ \ \mathcal{F}m \vdash fm \longrightarrow \mathcal{F}(Sm) \vdash f(Sm). \end{cases}$$

The first problem is already solved, since $\mathcal{F}0$ conv a. The second can be rewritten: (since we can replace $\mathcal{F}(Sm)$ by $gm(\mathcal{F}m)$, which conv $T'_3(\mathcal{F}m)$)

F: (m) $\mathcal{F}m \vdash fm$ $\longrightarrow$ $T'_3(\mathcal{F}m) \vdash f(Sm)$,

$$\text{Red.}\begin{cases} F\colon\ (m) \longrightarrow T' \vdash fm \supset f(Sm), \\ F\colon\ (m)\ \ \mathcal{F}m \vdash fm,\ \ T' \vdash fm \supset f(Sm) \longrightarrow T'_3(\mathcal{F}m) \vdash f(Sm); \end{cases}$$

the first problem reduces, by Lemma 0, to (*); the second is solved by Lemma 6 ($\mathcal{F}m$ is valid for F, and hence for fm, since a is valid for F and gm is valid for $F\supset F$).

This completes Rule (δ), and the proof of the main theorem justifying Heyting Arithmetic.∎

Chapter 3: Infinitary mathematics

§3.0 Introduction

The question for this chapter is: how is classical mathematics possible? Classical mathematics looks like an 'infinitised' version of finitary reasoning; in developing it mathematicians have extrapolated their finite abilities into the infinite. How do they get away with it? That is, how is it that, at the cost of jettisoning a few principles like 'the whole is greater than the part' and 'every property defines a class', they are able to develop an apparently consistent theory satisfying the remainder of our intuitions about quantities and collections? This is the problem of meeting Criterion (γ) of §0.1.

One possible answer is that infinities are all around us in the physical world and so it must be alright for mathematics to talk about them. I shall cast doubt on this in §3.1.

In §0.3 I suggested that classical mathematics was a distorted version of what I call infinitary mathematics, which is what one gets if one drops Assumption (γ) of §0.3 and assumes the ability to complete certain infinitistic computations. In §3.2 I shall show that mathematics under this kind of assumption looks much like finitary mathematics and does not produce anything as strong as analysis; this undermines the classical position further in that it shows that (part of) the underlying motivation for infinitistic arguments (the feeling that our inability to count up to ω is a 'mere medical accident', (cf Benacerraf and Putnam [0] p.17)), when developed carefully, does not lead to full classical mathematics.

In §§3.3,4 I shall approach analysis from a different angle (not involving denying Assumption (γ)), and show how it can be given a reasonable interpretation.

In §3.5 I indicate how far this approach can be generalised to the rest of mathematics.

§3.1 Do physical infinities exist?

Criterion (ε) of §0.1 says that a foundational system should explain how and why mathematics is applicable to science. Some people see this question as crucial: classical analysis is essential to physics, it is argued, so there must be something in it, and it is the job of mathematical philosophy to justify it. There is even the extreme formalist view that mathematics consists of formal systems which are entirely justified by the decidability of their axioms and rules and their empirical usefulness. Certainly one can define an arbitrary formal system by choosing an alphabet, a decidable class of 'well-formed formulae', a decidable subclass of 'axioms', and decidable 'rules'; and then 'apply' it by defining a mapping from well-formed formulae to experimental predictions; and empirically 'confirm' it by generating a large number of 'theorems' and testing the corresponding experimental predictions. But to accept this as a complete account of mathematics is to reduce it to a form of divination. Classical mathematics is an infinitised version of finitary reasoning, developed largely by mathematicians who thought they were literally reasoning about infinite systems: why should formal systems so obtained be any better at prediction than an arbitrary formal system (which may not remotely resemble a reasoning process), and why should formal systems be any better than reading tea-leaves? Suppose Peano Arithmetic applies successfully to two physical systems, but they disagree about the truth-value of (the prediction corresponding to) Goldbach's conjecture, say: what becomes of the claim that they are both actually infinite systems? Many branches of mathematics were developed in the absence of physical applications: some later proved applicable and others remain unapplied, yet mathematicians are still able to distinguish between valid and invalid proofs, and significant and insignificant concepts.

Part of the problem is that we do not know why the universe is explicable or predictable at all: but it is a sound heuristic principle that any concept that proves useful in science probably can be made into clear and coherent mathematics.

So: are real numbers useful in science? Certainly they are used in science, because they are the only mathematical objects most scientists know about (apart from vectors, tensors, etc, built up out of reals). The view that mathematics is 'the science of quantity' has led to an assumption that an argument is only mathematical, and hence only really scientific, if it contains a lot of symbols denoting real numbers; this attitude has distorted twentieth-century psychology, particularly, and may be beginning to have the same effect on physics. General considerations of quantum gravity suggest that space-time is not a manifold on the smallest scale, yet physicists still think perturbatively of a manifold with space-time fluctuations superimposed, when a more radical approach seems required. The real continuum was originally inspired by the assumption that space-time is 'in principle' infinitely divisible; once this is doubted the motivation for using real numbers disappears.

When infinities occur in physics they are generally indistinguishable from sufficiently large finite quantities; eg it is difficult to tell whether the universe is infinite or simply very large. In other words, the functions mapping reality to experimental observations are continuous at all infinities. If this is always true then perhaps it would be better to model reality with 'sufficiently large' finite (or possibly nonstandard) quantities, or some other way of combining the convenient aspects of finiteness and infinity. This continuity principle is not logically necessary, and it does seem suspicious that infiniteness is unobservable. Until someone discovers a suitable discontinuity, or a representation of space-time not based on real numbers, I regard the question of the necessity of classical mathematics for science as open.

§3.2 Infinite time-experience

The weakest form of infinity assumption is ω-infinitism, which states that finitary iterations, such as computing (fxpt Φ v), where ΦXx $\equiv$ if Cx then Rx else X(Hx), can always be completed. (Such a computation applies H repeatedly to v until a value x is obtained satisfying Cx, then the result is Rx.) This is represented in protologic by adding the iteration operator it_a (parametrised by a), defined by

$$\mathrm{it}_a\ \Phi\ v \equiv \begin{cases} \text{fxpt } \Phi\ v, & \text{if this is defined,} \\ a, & \text{if not } Cv, \text{ not } C(Hv), \text{ not } C(H^2v), \text{ etc;} \end{cases}$$

in other words, $\mathrm{it}_a\ \Phi$ applies H repeatedly to v, and if (not C) holds for each iterate it returns the result a. ('it_a' is only defined for Φ of the form (λXx. if Cx then Rx else X(Hx)); fxpt is defined for all Φ.) It follows from this description that if, say, $H^{27}v$ or $C(H^{27}v)$ is undefined then so are both fxpt Φ v and $\mathrm{it}_a\ \Phi$ v, since they both have to compute these expressions (assuming the computation doesn't halt before that stage). 'it_ω' describes a new primitive process of iterating something infinitely many times: it can therefore be applied to a Φ where C, R and H contain it_ω.

All the protological axioms and rules still hold: in addition we have protological rules governing it_ω: (where Φ is of the allowed form)

$$it_\omega\Phi x \quad \text{conv} \quad \text{if } it_{True}\Phi x = it_{False}\Phi x \text{ then fxpt } \Phi\, x \text{ else a,}$$

$$\Phi(it_\omega\Phi)x \quad \text{conv} \quad it_\omega\Phi x, \quad \text{for an object a} \neq \text{True},$$

$$\frac{(\underline{x},\underline{y})\ A(\underline{x}) \rightarrow \Phi A(\underline{x})}{(\underline{x},\underline{y})\ A(\underline{x}) \rightarrow it_{True}\Phi(\underline{x})}.$$

We can make finitary functions (ie ones not containing it_ω) total by replacing fxpt by it_ω: but this doesn't mean that all functions are now total, for the it_ω function allows us to define more complex recursions. We are particularly interested in recursions for which the fxpt rules apply: these are ones in which ΦXx is a conjunction of terms of the form Kx and X(Hx). In finitism this means, without loss of generality, $\Phi Xx \equiv$ if Cx then Rx else X(Hx); in ω-infinitism we also have the general form

$$\Phi Xx \ \equiv\ \text{if Cx then Rx else if } C'x \text{ then X(Hx) else}$$
$$it_{True}(\lambda Yy.\ \text{if } C''y \text{ then } R'y \text{ else } X(Ay)\ \&\ Y(H''y))(H'x),$$

which involves searching a tree (rather than merely iterating an operation) and calculating the conjunction of Ry or $R'y$ for various nodes y visited.

Clearly we can extend the system further by adding a new operation it_ω^1 which completes recursions of the above form; and likewise it_ω^2, it_ω^3, ...it_ω^ω, $it_\omega^{\omega+1}$, ... Each it_ω^α satisfies rules like the above rules for it_ω, but with a larger class of allowed Φ. In each case the overall shape of the universe of objects is roughly the same; the iteration functions are 'within' the system, like 'S' and '=', and, whatever infinitistic operations we are allowing ourselves, 'fxpt' enables us to define a computation which iterates them indefinitely: so that functions are in general partial, and totality is not 'decidable' in the system (by the usual diagonal argument).

A problem with these systems is that, from a finitary point of view, we can say so little about the size of the infinitistic universes; eg we cannot decide whether all objects are finitely definable in terms of the iteration operations.

It is informally clear that ω-infinitism allows us to justify Peano Arithmetic by treating arithmetic formulae as decidable, since infinite quantifiers $\forall x \in \mathbb{N}$, $\exists x \in \mathbb{N}$ can be simply *computed*. However, no infinity assumption of the kind I have discussed will take us as far as classical analysis: the most we can expect is predicative analysis. I have tried hard to form a coherent notion of 'computing' $\forall x \in \mathbb{R}\ A(x)$, and failed: one would need a time-experience in which one tests each $A(x)$ individually and halts if $A(x) \neq$ True. Lacking a well-ordering of $\mathbb{R}$, this cannot be construed as the completion of any simpler computation; and if there are several x for which $A(x) \neq$ True it is unclear at which x the computation should halt.

Thus, although the idea of an infinite time-experience completing a given type of computation makes sense, and can be used to interpret classical arithmetic, it does not seem capable of reaching classical analysis. Next section I shall try another approach.

§3.3 Choice sequences and analysis

Analysis (which I shall identify with second-order number theory) is motivated by a wish to talk about 'all sets, or predicates, of numbers' (equivalently, all real numbers) impredicatively, ie without having 'surveyed' or 'constructed' them all first; and even by such arguments to define new predicates of numbers which are supposed to be within the scope of the original quantification. But of course one can never *have constructed* infinitely many things: the problem of impredicativity arises even in number theory, when formulae quantifying over $\mathbb{N}$ are taken to be propositions and used in the induction axiom schema.

In my account of number theory I escaped the problem by denying that number-theoretic formulae were propositions. First I *defined* a notion of proving 'for all n, $\Phi(n)$, regardless of n', ie the protological predicate DT; then I used DT to give a meaning to formulae (ie to define $\vdash$); then I justified HA by showing that one can turn an HA derivation into a 'proof' of the formula derived. This works as an interpretation of arithmetic provided we accept that protologic has been clearly defined and is sound. In other words, we need to believe that there is an autonomous notion of 'arguing regardless of n' (not defined as an infinite conjunction of arguing for n = 0,1,2,...), and that any instantiation of such an argument (ie adding the information that in fact n = 27, say) preserves its validity. My philosophical position entitles me to affirm this (cf §§0.2,3): since all our reasoning about numbers involves abstraction, it must be legitimate to abstract from a further aspect, namely the value of n in an argument.

The same approach can be taken to analysis. We can think of an arbitrary predicate (of natural numbers) A and prove things about it simply on the assumption that $n \in A$ (or An, as I shall write it) is true or false if n is a number. Thus A is thought of as a 'black box' or 'oracle' delivering truth-values as output for numeric input; we know nothing of the internal workings of A, which might be an algorithm or a physical process or a combination of both.

As an example of A as a physical process consider A $\equiv$ 'the position of a particle on a unit line segment', with $An \equiv$ True if the n'th digit in the binary expansion of the position coordinate is 1, False if the digit is 0: this can be rephrased as saying that An is the proposition that the particle lies in a certain interval of length 2^{-n}, defined in terms of $A0,\ldots A(n-1)$, ie as a measurement. The system is a black box in that one can choose n, measure the position of the particle to accuracy 2^{-n}, and the result gives An. If space-time were infinitely divisible this would be a sensible way to introduce the concept of the space-time position of an event.

This view is very much like the orthodox intuitionistic theory of choice sequences (CS). However, CS has trouble with the question of identity of choice sequences: presumably $A = A$ is true regardless of A, but can $A = B$ ever be true, or what about $(\lambda n.\ \mathrm{not}(An)) = B$? The trouble is that choice sequences, or my 'predicates', are not purely extensional: they have also an intensional aspect, their 'identity' as sequences, which makes $A = A$ true. This disrupts the continuity axioms intuitionists wish to assert: the axioms have to be restricted by conditions that the bound choice sequence variables appearing in the axioms are not equal. This deals with problems arising from atomic formulae of the form $A = B$, but not ones of the form $\gamma A = \delta B$ (where γ and δ are continuous functionals).

CS ends up distinguishing between 'absolutely lawless' sequences, 'lawlike' sequences (given by a rule), and sequences 'in between' (eg, γA, for continuous γ and lawless A). This has the implausible consequence that if A is a (boolean-valued) lawless sequence then (λn. not(An)) is a non-lawless sequence. In the attempt to obtain a notion of choice sequence which allows 'restricted' sequences, is closed under continuous functionals, and satisfies useful continuity properties, CS has been developed into ever more elaborate systems [17], until (it seems to me) it collapses under its own weight.

The source of the trouble is that when one sequence is defined as the image of another under a continuous functional it is no longer an entirely 'black' box: part of its internal workings is known (more formally, the continuity axioms, which assume complete 'blackness', fail). Hence the distinction between lawless sequences and other non-lawlike ones (partially black boxes): see [17,p.16] and [2,p.421] for two incompatible accounts of this.

It seems more natural to me to allow $A = \gamma B$ (or, for that matter, $\gamma C = \delta D$) to be possibly true, for any choice sequence variables A,B,C,D, without any distinction between lawless, lawlike and other sequences. That is, γB can be *regarded as* a black box by disregarding our knowledge that it is obtained by applying γ to something, and the abstract extensionalised sequence so obtained may well be what A denotes; here A is lawlike relative to B, but all sequences on their own are equally lawless.

The disadvantage of this from the point of view of CS is that it would disrupt the continuity axioms. But this does not bother me as I will not need continuity arguments at all.

Thus it seems reasonable to modify the protologic by allowing two sorts of variables, one regarded as ranging over the 'objects' of §1.5 and the other regarded as names for arbitrary predicates of numbers (or, more conveniently, of objects), which I shall denote by lower and upper case metavariables respectively. We assume that terms like Px are meaningful. This gives a way of proving protological statements of the form 'for all P, ... , regardless of P', from which we might hope to develop analysis.

Recall the discussion of §0.2: statements of the form 'for all x, ...' are not directly meaningful unless they can be given a quasi-predictive meaning: so we cannot just set up the classical truth-definition for analysis, '$\forall P\ \Phi(P)$ is true iff $\Phi(P)$ is true for all P'. We can however say, 'an argument A(P) for $\Phi(P)$ is valid regardless of P', and we can instantiate P to anything with the required kind of meaning.

It is worth pausing to note why this approach does not generalise to allow full naive set theory, complete with Russell's paradox. The obvious way of getting set theory would be to allow set variables, $\alpha,\beta,\gamma,\ldots$, including ordinary objects within their scope, with all such terms as $\alpha\beta$ meaningful. But this doesn't work. Analysis works because we can instantiate a predicate variable P to any 'suitable' $\mathfrak{X}$: ie any $\mathfrak{X}$ for which $\mathfrak{X}x$ is meaningful for object variable x. But in the case of set theory a 'suitable' $\mathfrak{X}$ is one for which $\mathfrak{X}\beta$, $\mathfrak{X}\mathfrak{X}$, and $\beta\mathfrak{X}$ are meaningful for set variable β; ie one for which $\mathfrak{X}\mathfrak{Y}$, $\mathfrak{X}\mathfrak{X}$ and $\mathfrak{Y}\mathfrak{X}$ are meaningful for all 'suitable' $\mathfrak{Y}$. Thus, the definition of 'suitable' is circular. So we have no coherent idea of what set variables could legitimately be instantiated to (roughly paraphrased, we have no grasp of the meaning of 'set'): so set theory, unlike analysis, is not really an abstraction of anything.

When I say that Px is meaningful I mean that it is an incomplete statement, in the same sense in which an arithmetic formula is. This means that $y \vdash Px$ and 'y is valid for Px' are defined: P consists of two black boxes which, if they deliver an answer at all, give truth values for the two properties, given as input x and y. An argument about P can be instantiated by adding $P = (\lambda x.F)$, where F is a formula, or in any other way which will give values for $y \vdash Px$ and 'y is valid for Px'.

§3.4 Theorem justifying classical analysis

The protological sequent calculus for analysis is just as in §1.5, ignoring the distinction between the two sorts of variables. Even $(\lambda x.T)P$ conv $T(^P_x)$ is permitted: the sort distinction really only matters for free variables.

An <u>analytic formula</u> is: an atomic formula (a term), $A \wedge B$, $A \vee B$, $\neg A$, $A \supset B$, $A \Leftrightarrow B$, $\forall x\, A$, $\forall' P\, A$, $\exists x\, A$, $\exists' P\, A$, where A and B are formulae. Since I want to interpret *classical* analysis, I need a procedure similar to that used by Gödel to interpret classical number theory in intuitionistic number theory: I will take $\wedge$, $\supset$, $\forall$ and $\forall'$ as primitive and define quasi-classical $\vee$, $\exists$ and $\exists'$ in terms of them. Just as with Heyting Arithmetic, I regard formulae as disguised terms; they can be rewritten according to the following rules:

Every atomic formula A has an implicit $\neg\neg$ in front of it;

$A \vee B$ is short for $\neg(\neg A \wedge \neg B)$;

$\exists x\, A$ is short for $\neg\forall x\, \neg A$;

$\exists' P\ A$ is short for $\neg\forall' P\ \neg A$;

$\neg A$ is short for $A \supset \mathrm{False}$;

$A \Leftrightarrow B$ is short for $(A \supset B) \wedge (B \supset A)$;

$A \wedge B \equiv (\wedge\ A\ B)$;

$A \supset B \equiv (\supset \bar{A}\ \bar{B})$;

$\forall x\ A \equiv (\forall\ (\lambda x.A))$;

$\forall' P\ A \equiv (\forall'\ (\lambda P.A))$.

Thus, all formulae are rewritten as terms built up out of $\wedge$, $\supset$, $\forall$ and $\forall'$. In HA these symbols were simply arbitrary constants, serving as markers. Here it is more convenient to define them as the appropriate proof functions so that the definition of $\vdash$ will become $p \vdash F \equiv Fp$; ie to identify F with $(\lambda p.\ p \vdash F)$. This reflects the intuitionistic definition of the meaning of a formula as its test for proofhood. Specifically, $\wedge$, $\supset$, $\forall$ and $\forall'$ are defined by

$$\wedge\ A\ B\ (p,q) \equiv Ap\ \&\ Bq,$$

$$\supset \bar{A}\ \bar{B}\ ((g,h),d) \equiv d[(q) \rightarrow gq[q \vdash A \rightarrow hq \vdash B]],$$

$$\forall'\ f\ (g,d) \equiv \forall\ f\ (g,d) \equiv d[(x) \rightarrow gx \vdash fx].$$

An 'arithmetic' atomic formula T (ie not of the form Px) will still be represented as $\bar{T}$; Px will be represented as Px (both have $\neg\neg$ in front).

Thus, any formula is a function for testing possible proofs; considering it concretely as a program, the validity condition is inserted as a 'comment statement' in the program, which does not affect the execution of the program but can be extracted from it when needed. To complete the definitions of $\wedge$, $\supset$, $\forall$ and $\forall'$, we need to specify how each logical constant obtains the validity condition of a formula from the validity conditions of its components:

(p,q) is valid for $\wedge\ A\ B$ $\equiv$ p is valid for A and q is valid for B;

((g,h),d) is valid for $\supset \bar{A}\,\bar{B}$ $\equiv$ d is valid for dt, g is valid for

$A \Rightarrow$ dt, and h is valid for $A \Rightarrow B$;

(g,d) is valid for $\forall$ f $\equiv$ d is valid for dt

and gx is valid for fx for any x;

(g,d) is valid for $\forall'$ f $\equiv$ d is valid for dt

and gP is valid for fP for any P.

The difference between $\forall$ and $\forall'$ is in the validity condition: in arguing that gx is valid for fx for any x we can assume that for atomic formulae xy validity is trivial (anything is valid for them) and also that z $\vdash$ xy conv xy, whereas in arguing that gP is valid for fP for any P all we can assume is that 'valid for Py' and the function Py have a definite meaning. This is the only place where the difference between predicates and objects makes itself felt.

<u>Classical analysis</u> (CA) is defined as:

(i) HA with two sorts of variables, ie Axiom (x) splits into

(xa) $(\forall x\ A) \supset A(^{y}_{x})$, (xb) $(\forall' P\ A) \supset A(^{Q}_{P})$;

and Axiom (xi) and Rules (β) and (γ) split into (xia), (xib), (βa), (βb), (γa), (γb) similarly.

(ii) $A \vee \neg A$;

(iii) $\exists' P\ \forall x\ (Px \Leftrightarrow F)$ for any formula F not containing P.

<u>Theorem</u> (Interpretation of analysis) Any derivation of a CA theorem A can be transformed into a solution of $F(T^A)$: $\rightarrow T \vdash A$.

<u>Proof</u> As with the HA theorem, the procedure is to solve $F(T^A)$: $\rightarrow T \vdash A$ for each axiom schema A and to reduce $F(T^A)$: $\rightarrow T \vdash A$ to $F(P^B)$: $\rightarrow P \vdash B$ ... $F(Q^C)$: $\rightarrow Q \vdash C$ for each rule of inference $\frac{B \ldots C}{A}$.

For the HA part, note that the $\vee, \exists, \exists'$ axioms and rules (ie Axioms (v), (vi), (vii), (xia), (xib) and Rules (γa), (γb)) are redundant: they follow from the rest of the predicate calculus, bearing in mind the definitions of $\vee$, $\exists$ and $\exists'$ (to show this, first prove that $\neg\neg A \supset A$ is a theorem of the rest of the predicate calculus, by induction on the structure of A). $A \vee \neg A$ is likewise redundant.

The remaining predicate calculus axioms and rules (Axioms (i)-(iv), (viii)-(xb), Rules (α), (βa), (βb)) and also Axiom (xxii) are proved exactly, word for word, as in the HA theorem; all the things asserted to be valid in the HA proof are still valid in this context.

In the cases of Axioms (xiii)-(xxi) and Rule (δ) we have to be careful because of the implicit $\neg\neg$ in front of atomic formulae; but we can prove versions of them without the $\neg\neg$ exactly as in the HA proof and then, using $A \vee \neg A$, obtain by predicate calculus the versions with the $\neg\neg$.

The remaining HA axiom is (xii): $At \supset \exists x\ x{=}t$, which is handled as follows.

Firstly, (choosing a new variable y) we can solve

$F(V^{\neg y=t})\{R^{\forall x\ \neg x=t}\}$: $R \vdash \forall x\ \neg(x{=}t) \ \longrightarrow\ V \vdash \neg(y{=}t)$,

since $(\forall x\ \neg(x{=}t)) \supset \neg(y{=}t)$ is an axiom already proved; secondly, by Lemma 6 we can solve

$F\{R\}$: $O \vdash y{=}t,\ \ V \vdash \neg(y{=}t) \ \longrightarrow\ V_3O \vdash \mathrm{False}$.

Combining the two sequents with Cut and applying Lemma 0 solves

$F\{R\}$: $(y)\ \ y{=}t,\ \ R \vdash \forall x\ \neg(x{=}t) \ \longrightarrow\ V_3O \vdash \mathrm{False}$;

instantiating y to t and converting $t{=}t$ to $\overline{\mathrm{True}}\ t$ gives the sequent

$$\overline{\mathrm{True}}\ t,\ \ R \vdash \forall x\ \neg(x{=}t) \ \longrightarrow\ (\lambda y.V_3O)t \vdash \mathrm{False}.$$

Now put $P \equiv (\lambda y.V_3O)t$: since $Q \vdash At \ \longrightarrow\ \overline{\mathrm{True}}\ t$ is derivable this solves

$F(P^{False})\{Q^{At}, R^{\forall x\ \neg(x=t)}\}$: $Q \vdash At,\ \ R \vdash \forall x\ \neg(x=t) \ \longrightarrow\ P \vdash False,$

$\equiv$ (by Lemma 7 twice)

$F(P'^{At\supset((\forall x\ \neg(x=t))\supset False)})$: $\ \longrightarrow\ P' \vdash At \supset ((\forall x\ \neg(x=t)) \supset False)$:

but $At \supset ((\forall x\ \neg(x=t)) \supset False)$ is just $At \supset \exists x\ x=t$, as required.

All that remains is to prove the comprehension axiom, ie to solve

$F(T^{\exists' P\forall x(Px\Leftrightarrow F)})$: $\ \longrightarrow\ T \vdash \exists' P\ \forall x\ (Px \Leftrightarrow F)$.

(Note that Px has an implicit $\neg\neg$ in front of it, but since I have already proved $A \vee \neg A$ the $\neg\neg$ can be removed.)

First, I will find a proof of $Q=(\lambda x.F) \supset (Qx \supset F)$, ie solve

$F(T^{Q=(\lambda x.F)\supset(Qx\supset F)})$: $\ \longrightarrow\ T \vdash Q=(\lambda x.F) \supset (Qx \supset F)$,

$\equiv$ (Lemma 7 twice) $F(T'^{F})\{U^{Q=(\lambda x.F)}, V^{Qx}\}$: $U \vdash Q=(\lambda x.F),\ V \vdash Qx \longrightarrow T' \vdash F,$

Red. $F(T'^{F})\{U^{Q=(\lambda x.F)}, V^{Qx}\}$: $Q=(\lambda x.F),\ V \vdash F \longrightarrow T' \vdash F$;

this is solved by defining $T' \equiv$ (if $Q=(\lambda x.F)$ then V else V_0), where V_0 is a constant term trivially valid for F; for T' is clearly valid for F if V is valid for Qx, and the sequent is easily derivable.

Similarly, we can prove $Q=(\lambda x.F) \supset (F \supset Qx)$. Then by predicate calculus we can derive $Q=(\lambda x.F) \supset (Qx \Leftrightarrow F)$, and hence (by Rule ($\beta$a)) $Q=(\lambda x.F) \supset \forall x(Qx \Leftrightarrow F)$, and using Axiom (xib) we then derive $Q=(\lambda x.F) \supset \exists' P\forall x(Px \Leftrightarrow F)$. Since we have already proved the Theorem for all of CA except the comprehension axiom this means we can solve

$F(T^{Q=(\lambda x.F)\supset X})$: $\ \longrightarrow\ T \vdash Q=(\lambda x.F) \supset X$,

(abbreviating $\exists' P\forall x(Px \Leftrightarrow F)$ to X), which by Lemma 7 is equivalent to

$F(T'^{X})\{U^{Q=(\lambda x.F)}\}$: $U \vdash Q=(\lambda x.F) \ \longrightarrow\ T' \vdash X$.

Using a T' which solves this problem we can also solve (Lemma 0)

$F(U)$: $(Q)\ \ Q=(\lambda x.F) \ \longrightarrow\ T' \vdash X$,

and by the Instantiation Rule we derive immediately

$$(\lambda x.F)=(\lambda x.F) \;\rightarrow\; (\lambda Q.T')(\lambda x.F) \vdash X;$$

$(\lambda x.F)=(\lambda x.F)$ conv True, so, defining $T'' \equiv (\lambda Q.T')(\lambda x.F)$, we have solved $F(T''^{X})$: $\rightarrow\; T'' \vdash X$,

as required.

This completes the comprehension axiom, and hence the proof of the Theorem justifying Classical Analysis.■

From the point of view of classical proof theory this theorem is unsurprising: it uses the concept of validity, which includes well-foundedness of trees (which is Π^1_1 in the analytic hierarchy) and functions from well-founded trees to well-founded trees, etc. A classical proof theorist would probably want to formalise validity and express the theorem in the form 'analysis is consistent relative to formal validity theory'. However, I am not a classical proof theorist: I intend the theorem as an 'interpretation' of analysis in the everyday sense, ie as a systematic method of assigning meaning to it. Chapter 0 was intended as an account of mathematical activity, not as mere 'motivation' or verbal ornamentation for the formal system I eventually introduce in §1.5. So I am entitled to say, not 'analysis is consistent relative to ...', but 'analysis makes sense, when looked at in the right way, and is therefore consistent'.

§3.5 Set theory and general axiomatic theories

Clearly the procedure of the previous section for obtaining analysis can be iterated to obtain the theory of functions of real numbers, functions of real functions, and so on; and perhaps even the limit step, to allow us to talk about 'all functions of finite type'. This suggests that it may be possible to interpret a form of set theory, defining a set as a possible range for a variable. The difficulty here is with referring to the whole universe. ZF set theory, which allows us to quantify over the universe of sets but not to treat the range of quantification as an object, seems to me more than ever implausible as a solution to the problem of the 'whole universe'. I still have no explanation of why ZF is apparently consistent, except that few people have looked very hard for a contradiction where one is most likely to be found, in the mixture of static and generative ideas in the impredicative use of the replacement and separation axiom schemata.

I have justified arithmetic and analysis without being forced by the logic of my approach to continue to a vast and useless set hierarchy: this is an advantage of my system (by Criterion (δ), §0.1).

So far I have ignored general axiomatic theories such as group theory or general topology, in which one deals with an arbitrary domain X with some extra structure. This is easily amenable to my approach. One simply introduces in the protologic variables x,y,z,... regarded as ranging over X, and perhaps variables S,... ranging over subsets of X, and assumes that '$x \bullet y = z$' or 'S is open' is meaningful; this protologic is justified by the fact that it can be instantiated by taking for X any constructed domain with the required extra structure. One defines formulae as usual, and proceeds to prove formulae of the form '(axioms of the theory) $\supset$ (theorem of the theory)'. This is closer to the spirit of a general axiomatic theory than is the set-theoretic treatment, which only proves theorems for all sets X, whereas surely they should hold when X is a proper class or any collection whatever.

References

[0] Benacerraf, P. & Putnam, H. - *Philosophy of mathematics* - Prentice-Hall, New Jersey, 1964 (introduction).

[1] Bishop, E. - *Mathematics as a numerical language* - in [8], pp 53-71.

[2] Dummett, M. - *Elements of intuitionism* - Oxford University Press, 1977.

[3] Gödel, K. - *Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes* - Dialectica **12** (1958) pp 280-287.

[4] Goodman, N.D. - *A theory of constructions equivalent to arithmetic* - in [8], pp 101-120.

[5] Goodman, N.D. - *The arithmetic theory of constructions* - Cambridge Summer School in Mathematical Logic (Cambridge, England, 1971), (eds A.R.D.Mathias & H.Rogers), Lecture Notes in Mathematics **337**, Springer, Berlin, 1973, pp 274-298.

[6] Goodman, N.D. - *The faithfulness of the interpretation of arithmetic in the theory of constructions* - J.Symb.Logic **38** (1973), pp 453-459.

[7] Hallett, M. - *Cantorian set theory and limitation of size* - Oxford University Press, 1984.

[8] Kino, A., Myhill, J., & Vesley, R.E. (eds) - *Intuitionism and proof theory* - North-Holland, Amsterdam, 1970.

[9] Kleene, S.C. - *On the interpretation of intuitionistic number theory* - J.Symb.Logic **10** (1945) pp 109-124.

[10] Kolmogorov, A.N. - *Zer Deutung der intuitionistischen Logik* - Mathematische Zeitschrift **35** (1932) pp 58-65.

[11] Kreisel, G. - *Foundations of intuitionistic logic* - in *Logic, methodology and philosophy of science* (eds E.Nagel, P.Suppes, & A.Tarski), Stanford University Press, 1962, pp 198-210.

[12] Kreisel, G. - *Mathematical logic* - in *Lectures on modern mathematics*, Vol III (ed. T.L.Saaty), Wiley, New York, 1965, pp 95-195.

[13] Loeckx, J. & Sieber, K. - *The foundations of program verification* - Wiley-Teubner series in computer science, Stuttgart, 1984.

[14] Martin-Löf, P. - *An intuitionistic theory of types: predicative part* - in *Logic Colloquium 1973* (Bristol) (eds H.E.Rose & J.C.Shepherdson), North-Holland, Amsterdam, 1975, pp 73-118.

[15] Parikh, R. - *Existence and feasibility in arithmetic* - J.Symb.Logic 36 (1971) pp 494-508.

[16] Sundholm, G. - *Constructions, proof and the meaning of logical constants* - J.Phil.Logic 12 (1983) pp 151-172.

[17] Troelstra, A.S. - *Choice sequences* - Oxford University Press, 1977.

[18] van Heijenoort, J. - *From Frege to Gödel - a source-book in mathematical logic 1879-1931* - Harvard University Press, Cambridge, 1967.

[19] Weinstein, S. - *The intended interpretation of intuitionistic logic* - J.Phil.Logic 12 (1983) pp 261-270.

[20] Yessenin-Volpin, A.S. - *The ultra-intuitionistic criticism and the antitraditional program for foundations of mathematics* - in [8], pp 3-45.